Py-CoilGen

Generated by Doxygen 1.9.4

Chapter 1

README

1.0.1 Python Implementation

This project is the python implementation of the matlab project: CoilGen (https://github.com/Philipp-MR/CoilGen).

The python project reprogrammed the functionalitys starting in the Heard file: `CoilGen\CoilGen.m`. When the functionality is fine the stucture and the code itself was cleaned up and changes that do not change the results were made.

State 2022-11-07 working was stopped in calcCountoursByTriangularPotentialCuts.py (there is a TODO with further explaintation). Until there in a testcase the results are similar to the ones in the matlab implementation. The testcase in Matlab could be found in: `CoilGen\Examples\ygradient_coil.m`.

1.0.2 Testing with Pytest

There is a testcase implemented in testCase.py, on which it could be evaluated if the program still does what it should do. The results there are checked against the ones of the same case in the matlab project. For testing pytest is used.

1. If not already installed run: pip install -U pytest
2. To run the tests, navigate to the Py_Coilgen folder and run: pytest

1.0.3 Citation

For citation of this work, please refer to the following publication: https://onlinelibrary.wiley.com/doi/10.1002/mrm.29294 https://doi.org/10.1002/mrm.29294

(back to top)

Chapter 2

Namespace Index

2.1 Package List

Here are the packages with brief descriptions (if available):

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

Chapter 6

Namespace Documentation

6.1 calcContoursByTriangularPotentialCuts Namespace Reference

Functions

- def calcContoursByTriangluarPotentialCuts (mesh, potentialLevelList, streamFunction)
- def getuvCutPoints (mesh, innerEdges, edgeLength, potentialCutCriteria, cutPointDistanceToEdgeNode)
- def getRawUnsortedPoints (potentialLevelList, potentialSortedCutPoints)
- def getPotentialSortedCutPoints (potentialLevelList, uCutPoint, vCutPoint)
- def getCutDistancesToEdgeNodes (edgeNodePotentials, potentialLevelList, edgeLength)
- def getPotentialCutCriteria (edgeNodePotentials, potentialLevelList)
- def getRawUnarrangedLoops (rawUnsortedPoints, innerEdges, innerEdgeOpposedNode)
- def checkIfPositionsElementIdenticalWithFirstList (allCurrentEdges, allCurrentOpposedNodes, check↩ Position)
- def getEdgeOpposedNode (mesh, innerEdgesTrianglesInds, innerEdges)
- def removeTwoElementsFromArray (array, Element1, Element2)
- def getInnerEdegTriangleNodes (innerEdgesTrianglesInds, mesh)
- def getInnerEdges (mesh)
- def getNumAttachedTriangles (edgeAttachedTriangles)
- def getEdgeAttachedTriangles (edges, mesh)
- def getEdges (mesh)

6.1.1 Function Documentation

6.1.1.1 calcContoursByTriangluarPotentialCuts()

```
def calcContoursByTriangularPotentialCuts.calcContoursByTriangluarPotentialCuts (
              mesh,
              potentialLevelList,
              streamFunction )
```

Definition at line 4 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.2 checkIfPositionsElementIdenticalWithFirstList()

```
def calcContoursByTriangularPotentialCuts.checkIfPositionsElementIdenticalWithFirstList (
              allCurrentEdges,
              allCurrentOpposedNodes,
              checkPosition )

returns lists of elements with booleans if checkPostions Element of first and secound Element are identical wi
```

Definition at line 142 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.3 getCutDistancesToEdgeNodes()

```
def calcContoursByTriangularPotentialCuts.getCutDistancesToEdgeNodes (
              edgeNodePotentials,
              potentialLevelList,
              edgeLength )

 returns the cutPointDistanceToEdgeNode
```

Definition at line 57 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.4 getEdgeAttachedTriangles()

```
def calcContoursByTriangularPotentialCuts.getEdgeAttachedTriangles (
              edges,
              mesh )

returns the attached triangles for each edge.
```

Definition at line 198 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.5 getEdgeOpposedNode()

```
def calcContoursByTriangularPotentialCuts.getEdgeOpposedNode (
              mesh,
              innerEdgesTrianglesInds,
              innerEdges )

returns the to the edge opposed node for each attached triangle
```

Definition at line 156 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.6 getEdges()

```
def calcContoursByTriangularPotentialCuts.getEdges (
              mesh )
```

returns all edges in the mesh.

Definition at line 209 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.7 getInnerEdegTriangleNodes()

```
def calcContoursByTriangularPotentialCuts.getInnerEdegTriangleNodes (
              innerEdgesTrianglesInds,
              mesh )
```

returns the Node Indices for the Triangles.

Definition at line 171 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.8 getInnerEdges()

```
def calcContoursByTriangularPotentialCuts.getInnerEdges (
              mesh )
```

returns the inner edges an the corresponding triangleInds.

Definition at line 178 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.9 getNumAttachedTriangles()

```
def calcContoursByTriangularPotentialCuts.getNumAttachedTriangles (
              edgeAttachedTriangles )
```

returns the number of attached triangles per edge.

Definition at line 191 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.10 getPotentialCutCriteria()

```
def calcContoursByTriangularPotentialCuts.getPotentialCutCriteria (
              edgeNodePotentials,
              potentialLevelList )

returns the PotentialCutCriteria
```

Definition at line 69 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.11 getPotentialSortedCutPoints()

```
def calcContoursByTriangularPotentialCuts.getPotentialSortedCutPoints (
              potentialLevelList,
              uCutPoint,
              vCutPoint )

returns potentialSortedCutPoints
```

Definition at line 46 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.12 getRawUnarrangedLoops()

```
def calcContoursByTriangularPotentialCuts.getRawUnarrangedLoops (
              rawUnsortedPoints,
              innerEdges,
              innerEdgeOpposedNode )

returns rawUnarrangedLoops
```

Definition at line 83 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.13 getRawUnsortedPoints()

```
def calcContoursByTriangularPotentialCuts.getRawUnsortedPoints (
              potentialLevelList,
              potentialSortedCutPoints )

returns rawUnsortedPoints
```

Definition at line 34 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.14 getuvCutPoints()

```
def calcContoursByTriangularPotentialCuts.getuvCutPoints (
              mesh,
              innerEdges,
              edgeLength,
              potentialCutCriteria,
              cutPointDistanceToEdgeNode )

returns ucutPoint and vCutPoint
```

Definition at line 24 of file calcContoursByTriangularPotentialCuts.py.

6.1.1.15 removeTwoElementsFromArray()

```
def calcContoursByTriangularPotentialCuts.removeTwoElementsFromArray (
              array,
              Element1,
              Element2 )

returns the array without the two elements.
```

Definition at line 167 of file calcContoursByTriangularPotentialCuts.py.

6.2 calcPotentialLevels Namespace Reference

Functions

- def calcPotentialLevels (streamFunction, numLevels, levelOffset)

6.2.1 Function Documentation

6.2.1.1 calcPotentialLevels()

```
def calcPotentialLevels.calcPotentialLevels (
              streamFunction,
              numLevels,
              levelOffset )

returns the contourStep and the potentialLevelList. more options need to be implemented, if there is more than
```

Definition at line 3 of file calcPotentialLevels.py.

6.3 defineTargetField Namespace Reference

Classes

- class TargetField
- class TargetFieldGiven

Functions

- def distanceBetweenPoints (point1, point2)

6.3.1 Function Documentation

6.3.1.1 distanceBetweenPoints()

```
def defineTargetField.distanceBetweenPoints (
              point1,
              point2 )

returns the distance between two given points
```

Definition at line 43 of file defineTargetField.py.

6.4 main Namespace Reference

Variables

- Test = Tester()
- string meshFile = "cylinder_radius500mm_length1500mm.stl"

  Input ##################.
- string targetMeshFile = "sphere_radius150mm.stl"
- int gaussOrder = 2
- int tikonovFac = 100
- float specificConductivityMaterial = 1.8000∗10∗∗-8
- float conducterThickness = 0.005
- float materialFactor = specificConductivityMaterial/conducterThickness
- int numLevels = 20
- float levelOffset = 0.2500
- Mesh = CylindricMeshGiven(meshFile)
- TargetSphere = TargetFieldGiven(targetMeshFile,1)
- sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
- resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
- bFieldGeneratedByOptSF
- streamFunction
- contourStep
- potentialLevelList
- contour = calcContoursByTriangluarPotentialCuts(Mesh,potentialLevelList,streamFunction)

6.4.1 Variable Documentation

6.4.1.1 bFieldGeneratedByOptSF

```
main.bFieldGeneratedByOptSF
```

Definition at line 43 of file main.py.

6.4.1.2 conducterThickness

```
float main.conducterThickness = 0.005
```

Definition at line 17 of file main.py.

6.4.1.3 contour

```
main.contour = calcContoursByTriangluarPotentialCuts(Mesh,potentialLevelList,streamFunction)
```

Definition at line 49 of file main.py.

6.4.1.4 contourStep

```
main.contourStep
```

Definition at line 46 of file main.py.

6.4.1.5 gaussOrder

```
int main.gaussOrder = 2
```

Definition at line 14 of file main.py.

6.4.1.6 levelOffset

```
float main.levelOffset = 0.2500
```

Definition at line 20 of file main.py.

6.4.1.7 materialFactor

```
float main.materialFactor = specificConductivityMaterial/conducterThickness
```

Definition at line 18 of file main.py.

6.4.1.8 Mesh

```
main.Mesh = CylindricMeshGiven(meshFile)
```

Definition at line 26 of file main.py.

6.4.1.9 meshFile

```
string main.meshFile = "cylinder_radius500mm_length1500mm.stl"
```

Input ##################.

Definition at line 12 of file main.py.

6.4.1.10 numLevels

```
int main.numLevels = 20
```

Definition at line 19 of file main.py.

6.4.1.11 potentialLevelList

```
main.potentialLevelList
```

Definition at line 46 of file main.py.

6.4.1.12 resistanceMatrix

```
main.resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
```

Definition at line 38 of file main.py.

6.4.1.13 sensitivityMatrix

```
main.sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
```

Definition at line 35 of file main.py.

6.4.1.14 specificConductivityMaterial

```
float main.specificConductivityMaterial = 1.8000*10**-8
```

Definition at line 16 of file main.py.

6.4.1.15 streamFunction

```
main.streamFunction
```

Definition at line 43 of file main.py.

6.4.1.16 targetMeshFile

```
string main.targetMeshFile = "sphere_radius150mm.stl"
```

Definition at line 13 of file main.py.

6.4.1.17 TargetSphere

```
main.TargetSphere = TargetFieldGiven(targetMeshFile,1)
```

Definition at line 31 of file main.py.

6.4.1.18 Test

```
main.Test = Tester()
```

Definition at line 9 of file main.py.

6.4.1.19 tikonovFac

```
int main.tikonovFac = 100
```

Definition at line 15 of file main.py.

6.5 readMesh Namespace Reference

Classes

- class CylindricMesh
- class CylindricMeshGiven

Functions

- def calculateNormal (vec)
- def updateList (edgeList, otheredge)
- def getMeshFromSTL (filename)
- def checkIfVecInVeclist (node, vecList)

6.5.1 Function Documentation

6.5.1.1 calculateNormal()

```
def readMesh.calculateNormal (
              vec )
```

```
returns the norm for a given 3d vector
```

Definition at line 7 of file readMesh.py.

6.5.1.2 checkIfVecInVeclist()

```
def readMesh.checkIfVecInVeclist (
              node,
              vecList )

returns Boolean if a 3 components vec is in a list of 3 component elements
```

Definition at line 359 of file readMesh.py.

6.5.1.3 getMeshFromSTL()

```
def readMesh.getMeshFromSTL (
              filename )

returns vertices and faces from given stl file meshes
```

Definition at line 335 of file readMesh.py.

6.5.1.4 updateList()

```
def readMesh.updateList (
              edgeList,
              otheredge )

returns the edgeList without otheredge
```

Definition at line 17 of file readMesh.py.

6.6 resistanceMatrix Namespace Reference

Functions

- def getResistanceMatrix (test, mesh, materialFactor)
- def formFinalResistanceMat (resistanceMatrix, materialFactor)
- def createPreviousResistanceMat (mesh, matElementsShouldGetValue)
- def getResistanceSumForDifferent (mesh, trianglesWithBothNodes, nodeInd1, nodeInd2)
- def getResistanceSumForSame (mesh, nodeInd1)
- def calculateArea (Point1, Point2, Point3)
- def calculateCurrent (Point1, Point2, Point3)
- def getMatElementShouldGetValue (mesh)
- def getPartnerElement (triangle, trianglesWithBothNodes, nodeInd2)
- def elementInArray (array, value)
- def elementInAandB (a, b)
- def compareMultipleElementsBoolean (elements, testelements)
- def getSpatialDistancesMatrix (mesh)
- def getNeighbourhoodMatrix (mesh)
- def compareMultipleElementsBooleanTest ()

6.6.1 Function Documentation

6.6.1.1 calculateArea()

```
def resistanceMatrix.calculateArea (
              Point1,
              Point2,
              Point3 )

returns the area of a triangle with 3 given Points
```

Definition at line 52 of file resistanceMatrix.py.

6.6.1.2 calculateCurrent()

```
def resistanceMatrix.calculateCurrent (
              Point1,
              Point2,
              Point3 )

returns the current of a triangle with 3 given Points
```

Definition at line 56 of file resistanceMatrix.py.

6.6.1.3 compareMultipleElementsBoolean()

```
def resistanceMatrix.compareMultipleElementsBoolean (
              elements,
              testelements )

returns a array with boolean elements in the length of elements, True if the value is in testelements False ot
```

Definition at line 94 of file resistanceMatrix.py.

6.6.1.4 compareMultipleElementsBooleanTest()

```
def resistanceMatrix.compareMultipleElementsBooleanTest ( )

Test function: should always be True
```

Definition at line 123 of file resistanceMatrix.py.

6.6.1.5 createPreviousResistanceMat()

```
def resistanceMatrix.createPreviousResistanceMat (
              mesh,
              matElementsShouldGetValue )

returns resistanceMatrix mit entries in matElementsShouldGetValue
```

Definition at line 17 of file resistanceMatrix.py.

6.6.1.6 elementInAandB()

```
def resistanceMatrix.elementInAandB (
              a,
              b )

returns a list with elements that are in a and b
```

Definition at line 87 of file resistanceMatrix.py.

6.6.1.7 elementInArray()

```
def resistanceMatrix.elementInArray (
              array,
              value )

returns a list with the elements the contain "value"
```

Definition at line 79 of file resistanceMatrix.py.

6.6.1.8 formFinalResistanceMat()

```
def resistanceMatrix.formFinalResistanceMat (
              resistanceMatrix,
              materialFactor )

returns resistanceMatrix in final form. added with its transposed and multiplied with materialFactor
```

Definition at line 12 of file resistanceMatrix.py.

6.6.1.9 getMatElementShouldGetValue()

```
def resistanceMatrix.getMatElementShouldGetValue (
              mesh )

returns a list with positions in the matix that should get a value (diagonal elements == same nodes and neighb
```

Definition at line 60 of file resistanceMatrix.py.

6.6.1.10 getNeighbourhoodMatrix()

```
def resistanceMatrix.getNeighbourhoodMatrix (
              mesh )

returns a boolean matrix with information if node i and node j are neighbours
```

Definition at line 112 of file resistanceMatrix.py.

6.6.1.11 getPartnerElement()

```
def resistanceMatrix.getPartnerElement (
              triangle,
              trianglesWithBothNodes,
              nodeInd2 )

returns the specified triangle both nodes are in from the oneRingList of the other one node. Important because
```

Definition at line 68 of file resistanceMatrix.py.

6.6.1.12 getResistanceMatrix()

```
def resistanceMatrix.getResistanceMatrix (
              test,
              mesh,
              materialFactor )

returns the resistance Matrix for the given mesh
```

Definition at line 4 of file resistanceMatrix.py.

6.6.1.13 getResistanceSumForDifferent()

```
def resistanceMatrix.getResistanceSumForDifferent (
              mesh,
              trianglesWithBothNodes,
              nodeInd1,
              nodeInd2 )

returns resistanceSum for the Case node1 and node2 are different
```

Definition at line 32 of file resistanceMatrix.py.

6.6.1.14 getResistanceSumForSame()

```
def resistanceMatrix.getResistanceSumForSame (
              mesh,
              nodeInd1 )

returns resistanceSum for the Case node1 and node2 are the same
```

Definition at line 43 of file resistanceMatrix.py.

6.6.1.15 getSpatialDistancesMatrix()

```
def resistanceMatrix.getSpatialDistancesMatrix (
              mesh )

returns a matrix containing the spatial distance between node i and node j
```

Definition at line 104 of file resistanceMatrix.py.

6.7 sensitivityMatrix Namespace Reference

Functions

- def getSensitivityMatrix (test, mesh, target, n)
- def calcSensitivityMat (mesh, biotSavatCoeff, target, u, v, gaussWeight)
- def gaussLegendreIntegrationPointsTriangle (test, n)
- def calcWeightsGauss (n)

6.7.1 Function Documentation

6.7.1.1 calcSensitivityMat()

```
def sensitivityMatrix.calcSensitivityMat (
              mesh,
              biotSavatCoeff,
              target,
              u,
              v,
              gaussWeight )

returns the calculated sensitivityMatrix
```

Definition at line 12 of file sensitivityMatrix.py.

6.7.1.2 calcWeightsGauss()

```
def sensitivityMatrix.calcWeightsGauss (
              n )

returns the abscissa and the weights for a Gauss-Legendre quadrature
```

Definition at line 56 of file sensitivityMatrix.py.

6.7.1.3 gaussLegendreIntegrationPointsTriangle()

```
def sensitivityMatrix.gaussLegendreIntegrationPointsTriangle (
              test,
              n )

returns the weights and the test point for the gauss legendre
```

Definition at line 43 of file sensitivityMatrix.py.

6.7.1.4 getSensitivityMatrix()

```
def sensitivityMatrix.getSensitivityMatrix (
              test,
              mesh,
              target,
              n )

returns the sensitivity Matrix for the mesh
```

Definition at line 5 of file sensitivityMatrix.py.

6.8 showSTLMesh Namespace Reference

Classes

- class CylindricMeshGiven

Functions

- def getMesh (filename)

Variables

- givenMesh = CylindricMeshGiven()
- figure = pyplot.figure()
- axes = mplot3d.Axes3D(figure)
- your_mesh = mesh.Mesh.from_file('cylinder_radius500mm_length1500mm.stl')
- scale = your_mesh.points.flatten()

6.8.1 Function Documentation

6.8.1.1 getMesh()

```
def showSTLMesh.getMesh (
              filename )
```

Definition at line 6 of file showSTLMesh.py.

6.8.2 Variable Documentation

6.8.2.1 axes

```
showSTLMesh.axes = mplot3d.Axes3D(figure)
```

Definition at line 23 of file showSTLMesh.py.

6.8.2.2 figure

```
showSTLMesh.figure = pyplot.figure()
```

Definition at line 22 of file showSTLMesh.py.

6.8.2.3 givenMesh

```
showSTLMesh.givenMesh = CylindricMeshGiven()
```

Definition at line 17 of file showSTLMesh.py.

6.8.2.4 scale

```
showSTLMesh.scale = your_mesh.points.flatten()
```

Definition at line 35 of file showSTLMesh.py.

6.8.2.5 your_mesh

```
showSTLMesh.your_mesh = mesh.Mesh.from_file('cylinder_radius500mm_length1500mm.stl')
```

Definition at line 28 of file showSTLMesh.py.

6.9 streamFunctionOptimization Namespace Reference

Functions

- def streamFunctionOptimization (test, mesh, target, sensitivityMatrix, resistanceMatrix, tikonovFactor)
- def applyTikonovRegularisation (tikonovFactor, redSenMat, redResMat, target)
- def reduceMatricesForBoundaryNodes (mesh, matToRed, zeroFlag)
- def rearrangeReducedMat (mesh, dimToRed, reducedMat)
- def getBoundaryDetails (mesh)
- def getReducedMat (mesh, dimToRed, reducedMat, zeroFlag)
- def getNotBoundaryNodes (mesh)
- def getDimToRed (matToRed)
- def getNumNodesPerBoundary (mesh)
- def reexpandSteamFunctionForBoundaryNodes (mesh, reducedSF, boundaryNodes, isNotBoundaryNode, zeroFlag)
- def updateMeshCurrentDensityMeshFaces (mesh, optStreamFkt)

6.9.1 Function Documentation

6.9.1.1 applyTikonovRegularisation()

```
def streamFunctionOptimization.applyTikonovRegularisation (
              tikonovFactor,
              redSenMat,
              redResMat,
              target )
```

returns the StreamFunction for the reduced Matrix with applied trikonov regularisation.

Definition at line 24 of file streamFunctionOptimization.py.

6.9.1.2 getBoundaryDetails()

```
def streamFunctionOptimization.getBoundaryDetails (
              mesh )
```

returns numNodesPerBoundary,notBoundaryNodes,boundaryNodes for the given mesh.

Definition at line 66 of file streamFunctionOptimization.py.

6.9.1.3 getDimToRed()

```
def streamFunctionOptimization.getDimToRed (
              matToRed )
```

returns boolean values which dimension should be reduced

Definition at line 101 of file streamFunctionOptimization.py.

6.9.1.4 getNotBoundaryNodes()

```
def streamFunctionOptimization.getNotBoundaryNodes (
              mesh )
```

returns no-boundary nodes of a given mesh

Definition at line 92 of file streamFunctionOptimization.py.

6.9.1.5 getNumNodesPerBoundary()

```
def streamFunctionOptimization.getNumNodesPerBoundary (
              mesh )

returns the number of nodes for every boundary
```

Definition at line 109 of file streamFunctionOptimization.py.

6.9.1.6 getReducedMat()

```
def streamFunctionOptimization.getReducedMat (
              mesh,
              dimToRed,
              reducedMat,
              zeroFlag )

returns the reduced matrix
```

Definition at line 73 of file streamFunctionOptimization.py.

6.9.1.7 rearrangeReducedMat()

```
def streamFunctionOptimization.rearrangeReducedMat (
              mesh,
              dimToRed,
              reducedMat )

rearranges the reduced matrix. returns: reducedMat,boundaryNodes,notBoundaryNodes
```

Definition at line 41 of file streamFunctionOptimization.py.

6.9.1.8 reduceMatricesForBoundaryNodes()

```
def streamFunctionOptimization.reduceMatricesForBoundaryNodes (
              mesh,
              matToRed,
              zeroFlag )

returns the for the boundary nodes reduced matrix
```

Definition at line 31 of file streamFunctionOptimization.py.

6.9.1.9 reexpandSteamFunctionForBoundaryNodes()

```
def streamFunctionOptimization.reexpandSteamFunctionForBoundaryNodes (
              mesh,
              reducedSF,
              boundaryNodes,
              isNotBoundaryNode,
              zeroFlag )

reexpand Streamfunction to all nodes, because the nodes of the first boundary have a potential of zero
```

Definition at line 116 of file streamFunctionOptimization.py.

6.9.1.10 streamFunctionOptimization()

```
def streamFunctionOptimization.streamFunctionOptimization (
              test,
              mesh,
              target,
              sensitivityMatrix,
              resistanceMatrix,
              tikonovFactor )

returns the magnetic field generated by the optimized stream function. Version for only one coil part, else  a
```

Definition at line 4 of file streamFunctionOptimization.py.

6.9.1.11 updateMeshCurrentDensityMeshFaces()

```
def streamFunctionOptimization.updateMeshCurrentDensityMeshFaces (
              mesh,
              optStreamFkt )

updates the current density of the Faces in the mesh
```

Definition at line 128 of file streamFunctionOptimization.py.

6.10 testCase Namespace Reference

Functions

- def test_finalSF ()
- def test_bFieldGeneratedByOptSF ()
- def test_reducedSF ()
- def test_gaußLegendre ()
- def test_WeightsGauss ()
- def test_matElementsShouldGetValue ()
- def main ()

Variables

- string meshFile = "cylinder_radius500mm_length1500mm.stl"
- string targetMeshFile = "sphere_radius150mm.stl"
- int gaussOrder = 2
- int tikonovFac = 100
- float specificConductivityMaterial = 1.8000∗10∗∗-8
- float conducterThickness = 0.005
- float materialFactor = specificConductivityMaterial/conducterThickness
- Test = Tester()
- Mesh = CylindricMeshGiven(meshFile)
- TargetSphere = TargetFieldGiven(targetMeshFile,1)
- sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
- resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
- BField
- SFOpt
- list matElementsShouldGetValueCorrect = [0.0000000000, 1.0000000000, 2.0000000000, 3.0000000000, 4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000, 9.0000000000, 10.↩ 0000000000, 11.0000000000, 12.0000000000, 13.0000000000, 14.0000000000, 15.0000000000, 16.↩ 0000000000, 17.0000000000, 18.0000000000, 19.0000000000, 20.0000000000, 21.0000000000, 22.↩ 0000000000, 23.0000000000, 24.0000000000, 25.0000000000, 26.0000000000, 27.0000000000, 28.↩ 0000000000, 29.0000000000, 30.0000000000, 31.0000000000, 32.0000000000, 33.0000000000, 34.↩ 0000000000, 35.0000000000, 36.0000000000, 37.0000000000, 38.0000000000, 39.0000000000, 40.↩ 0000000000, 41.0000000000, 42.0000000000, 43.0000000000, 44.0000000000, 45.0000000000, 46.↩ 0000000000, 47.0000000000, 48.0000000000, 49.0000000000, 50.0000000000, 51.0000000000, 52.↩ 0000000000, 53.0000000000, 54.0000000000, 55.0000000000, 56.0000000000, 57.0000000000, 58.↩ 0000000000, 59.0000000000, 60.0000000000, 61.0000000000, 62.0000000000, 63.0000000000, 64.↩ 0000000000, 65.0000000000, 66.0000000000, 67.0000000000, 68.0000000000, 69.0000000000, 70.↩ 0000000000, 71.0000000000, 72.0000000000, 73.0000000000, 74.0000000000, 75.0000000000, 76.↩ 0000000000, 77.0000000000, 78.0000000000, 79.0000000000, 80.0000000000, 81.0000000000, 82.↩ 0000000000, 83.0000000000, 84.0000000000, 85.0000000000, 86.0000000000, 87.0000000000, 88.↩ 0000000000, 89.0000000000, 90.0000000000, 91.0000000000, 92.0000000000, 93.0000000000, 94.↩ 0000000000, 95.0000000000, 96.0000000000, 97.0000000000, 98.0000000000, 99.0000000000, 100.↩ 0000000000, 101.0000000000, 102.0000000000, 103.0000000000, 104.0000000000, 105.0000000000, 106.0000000000, 107.0000000000, 108.0000000000, 109.0000000000, 110.0000000000, 111.↩ 0000000000, 112.0000000000, 113.0000000000, 114.0000000000, 115.0000000000, 116.0000000000, 117.0000000000, 118.0000000000, 119.0000000000, 120.0000000000, 121.0000000000, 122.↩ 0000000000, 123.0000000000, 124.0000000000, 125.0000000000, 126.0000000000, 127.0000000000, 128.0000000000, 129.0000000000, 130.0000000000, 131.0000000000, 132.0000000000, 133.↩ 0000000000, 134.0000000000, 135.0000000000, 136.0000000000, 137.0000000000, 138.0000000000, 139.0000000000, 140.0000000000, 141.0000000000, 142.0000000000, 143.0000000000, 144.↩ 0000000000, 145.0000000000, 146.0000000000, 147.0000000000, 148.0000000000, 149.0000000000, 150.0000000000, 151.0000000000, 152.0000000000, 153.0000000000, 154.0000000000, 155.↩ 0000000000, 156.0000000000, 157.0000000000, 158.0000000000, 159.0000000000, 160.0000000000, 161.0000000000, 162.0000000000, 163.0000000000, 164.0000000000, 165.0000000000, 166.↩ 0000000000, 167.0000000000, 168.0000000000, 169.0000000000, 170.0000000000, 171.0000000000, 172.0000000000, 173.0000000000, 174.0000000000, 175.0000000000, 176.0000000000, 177.↩ 0000000000, 178.0000000000, 179.0000000000, 180.0000000000, 181.0000000000, 182.0000000000, 183.0000000000, 184.0000000000, 185.0000000000, 186.0000000000, 187.0000000000, 188.↩ 0000000000, 189.0000000000, 190.0000000000, 191.0000000000, 192.0000000000, 193.0000000000, 194.0000000000, 195.0000000000, 196.0000000000, 197.0000000000, 198.0000000000, 199.↩ 0000000000, 200.0000000000, 201.0000000000, 202.0000000000, 203.0000000000, 204.0000000000, 205.0000000000, 206.0000000000, 207.0000000000, 208.0000000000, 209.0000000000, 210.↩ 0000000000, 211.0000000000, 212.0000000000, 213.0000000000, 214.0000000000, 215.0000000000, 216.0000000000, 217.0000000000, 218.0000000000, 219.0000000000, 220.0000000000, 221.↩ 0000000000, 222.0000000000, 223.0000000000, 224.0000000000, 225.0000000000, 226.0000000000, 227.0000000000, 228.0000000000, 229.0000000000, 230.0000000000, 231.0000000000, 232.↩ 0000000000, 233.0000000000, 234.0000000000, 235.0000000000, 236.0000000000, 237.0000000000,

238.0000000000, 239.0000000000, 240.0000000000, 241.0000000000, 242.0000000000, 243.↩
0000000000, 244.0000000000, 245.0000000000, 246.0000000000, 247.0000000000, 248.0000000000,
249.0000000000, 250.0000000000, 251.0000000000, 252.0000000000, 253.0000000000, 254.↩
0000000000, 255.0000000000, 256.0000000000, 257.0000000000, 258.0000000000, 259.0000000000,
260.0000000000, 261.0000000000, 262.0000000000, 263.0000000000, 1.0000000000, 2.0000000000,
0.0000000000, 6.0000000000, 8.0000000000, 9.0000000000, 3.0000000000, 2.0000000000, 7.↩
0000000000, 3.0000000000, 1.0000000000, 5.0000000000, 4.0000000000, 1.0000000000, 9.0000000000,
4.0000000000, 2.0000000000, 12.0000000000, 10.0000000000, 18.0000000000, 12.0000000000, 2.↩
0000000000, 5.0000000000, 3.0000000000, 10.0000000000, 4.0000000000, 11.0000000000, 14.↩
0000000000, 8.0000000000, 15.0000000000, 7.0000000000, 1.0000000000, 14.0000000000, 6.↩
0000000000, 0.0000000000, 1.0000000000, 6.0000000000, 15.0000000000, 16.0000000000, 17.↩
0000000000, 1.0000000000, 23.0000000000, 9.0000000000, 12.0000000000, 17.0000000000, 1.↩
0000000000, 3.0000000000, 8.0000000000, 18.0000000000, 78.0000000000, 4.0000000000, 5.↩
0000000000, 81.0000000000, 77.0000000000, 11.0000000000, 77.0000000000, 10.0000000000, 18.↩
0000000000, 9.0000000000, 17.0000000000, 4.0000000000, 3.0000000000, 19.0000000000, 21.↩
0000000000, 7.0000000000, 14.0000000000, 15.0000000000, 22.0000000000, 21.0000000000, 13.↩
0000000000, 7.0000000000, 6.0000000000, 14.0000000000, 29.0000000000, 8.0000000000, 6.↩
0000000000, 22.0000000000, 23.0000000000, 8.0000000000, 30.0000000000, 17.0000000000, 23.↩
0000000000, 25.0000000000, 24.0000000000, 9.0000000000, 12.0000000000, 25.0000000000, 16.↩
0000000000, 27.0000000000, 8.0000000000, 19.0000000000, 26.0000000000, 12.0000000000, 10.↩
0000000000, 4.0000000000, 81.0000000000, 85.0000000000, 19.0000000000, 26.0000000000, 33.↩
0000000000, 27.0000000000, 12.0000000000, 18.0000000000, 17.0000000000, 21.0000000000, 13.↩
0000000000, 14.0000000000, 20.0000000000, 28.0000000000, 13.0000000000, 144.0000000000, 143.↩
0000000000, 22.0000000000, 15.0000000000, 28.0000000000, 14.0000000000, 35.0000000000, 29.↩
0000000000, 21.0000000000, 29.0000000000, 30.0000000000, 8.0000000000, 15.0000000000, 16.↩
0000000000, 37.0000000000, 30.0000000000, 31.0000000000, 25.0000000000, 16.0000000000, 32.↩
0000000000, 16.0000000000, 17.0000000000, 31.0000000000, 27.0000000000, 24.0000000000, 85.↩
0000000000, 18.0000000000, 19.0000000000, 33.0000000000, 90.0000000000, 33.0000000000, 32.↩
0000000000, 17.0000000000, 25.0000000000, 19.0000000000, 34.0000000000, 151.0000000000, 22.↩
0000000000, 35.0000000000, 147.0000000000, 144.0000000000, 21.0000000000, 30.0000000000, 35.↩
0000000000, 36.0000000000, 22.0000000000, 23.0000000000, 15.0000000000, 29.0000000000, 43.↩
0000000000, 37.0000000000, 23.0000000000, 24.0000000000, 36.0000000000, 16.0000000000, 37.↩
0000000000, 24.0000000000, 32.0000000000, 39.0000000000, 38.0000000000, 25.0000000000, 39.↩
0000000000, 41.0000000000, 31.0000000000, 25.0000000000, 34.0000000000, 27.0000000000, 19.↩
0000000000, 27.0000000000, 34.0000000000, 90.0000000000, 40.0000000000, 26.0000000000, 41.↩
0000000000, 40.0000000000, 47.0000000000, 32.0000000000, 27.0000000000, 33.0000000000, 36.↩
0000000000, 29.0000000000, 151.0000000000, 22.0000000000, 42.0000000000, 28.0000000000, 42.↩
0000000000, 29.0000000000, 35.0000000000, 30.0000000000, 43.0000000000, 43.0000000000, 44.↩
0000000000, 24.0000000000, 30.0000000000, 31.0000000000, 38.0000000000, 31.0000000000, 39.↩
0000000000, 37.0000000000, 46.0000000000, 45.0000000000, 44.0000000000, 46.0000000000, 48.↩
0000000000, 41.0000000000, 38.0000000000, 31.0000000000, 32.0000000000, 34.0000000000, 90.↩
0000000000, 33.0000000000, 96.0000000000, 47.0000000000, 103.0000000000, 34.0000000000, 54.↩
0000000000, 47.0000000000, 48.0000000000, 32.0000000000, 39.0000000000, 35.0000000000, 43.↩
0000000000, 156.0000000000, 36.0000000000, 162.0000000000, 49.0000000000, 151.0000000000, 50.↩
0000000000, 36.0000000000, 37.0000000000, 56.0000000000, 44.0000000000, 30.0000000000, 42.↩
0000000000, 49.0000000000, 51.0000000000, 38.0000000000, 43.0000000000, 57.0000000000, 45.↩
0000000000, 37.0000000000, 50.0000000000, 52.0000000000, 51.0000000000, 46.0000000000, 44.↩
0000000000, 38.0000000000, 39.0000000000, 38.0000000000, 52.0000000000, 53.0000000000, 45.↩
0000000000, 48.0000000000, 34.0000000000, 103.0000000000, 40.0000000000, 54.0000000000, 41.↩
0000000000, 39.0000000000, 53.0000000000, 55.0000000000, 41.0000000000, 54.0000000000, 46.↩
0000000000, 61.0000000000, 42.0000000000, 162.0000000000, 43.0000000000, 56.0000000000, 57.↩
0000000000, 63.0000000000, 56.0000000000, 44.0000000000, 43.0000000000, 45.0000000000, 44.↩
0000000000, 52.0000000000, 58.0000000000, 57.0000000000, 65.0000000000, 45.0000000000, 46.↩
0000000000, 51.0000000000, 60.0000000000, 59.0000000000, 58.0000000000, 53.0000000000, 46.↩
0000000000, 55.0000000000, 52.0000000000, 60.0000000000, 48.0000000000, 48.0000000000, 109.↩
0000000000, 61.0000000000, 47.0000000000, 103.0000000000, 115.0000000000, 41.0000000000, 60.↩
0000000000, 62.0000000000, 61.0000000000, 48.0000000000, 53.0000000000, 50.0000000000, 49.↩
0000000000, 43.0000000000, 63.0000000000, 169.0000000000, 162.0000000000, 58.0000000000, 51.↩

0000000000, 63.0000000000, 64.0000000000, 50.0000000000, 44.0000000000, 51.0000000000, 57.↩
0000000000, 65.0000000000, 52.0000000000, 64.0000000000, 67.0000000000, 72.0000000000, 52.↩
0000000000, 66.0000000000, 65.0000000000, 60.0000000000, 52.0000000000, 53.0000000000, 59.↩
0000000000, 67.0000000000, 62.0000000000, 55.0000000000, 121.0000000000, 68.0000000000, 62.↩
0000000000, 115.0000000000, 48.0000000000, 55.0000000000, 54.0000000000, 55.0000000000, 68.↩
0000000000, 61.0000000000, 60.0000000000, 69.0000000000, 67.0000000000, 64.0000000000, 56.↩
0000000000, 57.0000000000, 50.0000000000, 175.0000000000, 70.0000000000, 169.0000000000, 58.↩
0000000000, 57.0000000000, 71.0000000000, 70.0000000000, 63.0000000000, 65.0000000000, 52.↩
0000000000, 64.0000000000, 58.0000000000, 59.0000000000, 72.0000000000, 75.0000000000, 71.↩
0000000000, 59.0000000000, 72.0000000000, 66.0000000000, 62.0000000000, 59.0000000000, 60.↩
0000000000, 127.0000000000, 69.0000000000, 73.0000000000, 121.0000000000, 61.0000000000, 62.↩
0000000000, 68.0000000000, 67.0000000000, 62.0000000000, 63.0000000000, 71.0000000000, 64.↩
0000000000, 181.0000000000, 187.0000000000, 175.0000000000, 74.0000000000, 74.0000000000,
64.0000000000, 70.0000000000, 65.0000000000, 75.0000000000, 75.0000000000, 65.0000000000,
59.0000000000, 68.0000000000, 69.0000000000, 76.0000000000, 75.0000000000, 71.0000000000,
187.0000000000, 193.0000000000, 70.0000000000, 74.0000000000, 65.0000000000, 76.0000000000,
71.0000000000, 193.0000000000, 74.0000000000, 10.0000000000, 78.0000000000, 79.0000000000,
10.0000000000, 84.0000000000, 77.0000000000, 81.0000000000, 80.0000000000, 79.0000000000, 83.↩
0000000000, 80.0000000000, 82.0000000000, 78.0000000000, 79.0000000000, 82.0000000000, 84.↩
0000000000, 78.0000000000, 18.0000000000, 10.0000000000, 84.0000000000, 85.0000000000, 78.↩
0000000000, 88.0000000000, 86.0000000000, 80.0000000000, 83.0000000000, 87.0000000000, 84.↩
0000000000, 79.0000000000, 86.0000000000, 82.0000000000, 82.0000000000, 81.0000000000, 89.↩
0000000000, 88.0000000000, 80.0000000000, 85.0000000000, 78.0000000000, 81.0000000000, 18.↩
0000000000, 84.0000000000, 26.0000000000, 90.0000000000, 89.0000000000, 91.0000000000, 87.↩
0000000000, 92.0000000000, 82.0000000000, 93.0000000000, 82.0000000000, 91.0000000000, 88.↩
0000000000, 86.0000000000, 89.0000000000, 93.0000000000, 87.0000000000, 82.0000000000, 94.↩
0000000000, 84.0000000000, 90.0000000000, 84.0000000000, 88.0000000000, 95.0000000000, 85.↩
0000000000, 94.0000000000, 89.0000000000, 40.0000000000, 102.0000000000, 26.0000000000, 33.↩
0000000000, 95.0000000000, 85.0000000000, 96.0000000000, 93.0000000000, 92.0000000000, 97.↩
0000000000, 99.0000000000, 86.0000000000, 87.0000000000, 98.0000000000, 91.0000000000, 97.↩
0000000000, 87.0000000000, 99.0000000000, 100.0000000000, 91.0000000000, 94.0000000000, 88.↩
0000000000, 105.0000000000, 89.0000000000, 88.0000000000, 106.0000000000, 100.0000000000,
101.0000000000, 93.0000000000, 95.0000000000, 101.0000000000, 102.0000000000, 90.0000000000,
89.0000000000, 94.0000000000, 103.0000000000, 108.0000000000, 40.0000000000, 90.0000000000,
102.0000000000, 209.0000000000, 210.0000000000, 104.0000000000, 92.0000000000, 213.0000000000,
99.0000000000, 98.0000000000, 91.0000000000, 97.0000000000, 209.0000000000, 104.0000000000,
105.0000000000, 93.0000000000, 97.0000000000, 91.0000000000, 94.0000000000, 93.0000000000,
105.0000000000, 106.0000000000, 102.0000000000, 106.0000000000, 94.0000000000, 95.0000000000,
112.0000000000, 107.0000000000, 101.0000000000, 107.0000000000, 96.0000000000, 113.0000000000,
95.0000000000, 90.0000000000, 108.0000000000, 108.0000000000, 109.0000000000, 114.0000000000,
47.0000000000, 96.0000000000, 40.0000000000, 54.0000000000, 110.0000000000, 99.0000000000,
213.0000000000, 97.0000000000, 219.0000000000, 105.0000000000, 100.0000000000, 106.0000000000,
99.0000000000, 104.0000000000, 93.0000000000, 110.0000000000, 116.0000000000, 111.0000000000,
105.0000000000, 100.0000000000, 112.0000000000, 111.0000000000, 117.0000000000, 94.0000000000,
101.0000000000, 113.0000000000, 118.0000000000, 102.0000000000, 101.0000000000, 112.↩
0000000000, 103.0000000000, 113.0000000000, 114.0000000000, 102.0000000000, 96.0000000000,
114.0000000000, 115.0000000000, 103.0000000000, 54.0000000000, 120.0000000000, 116.0000000000,
219.0000000000, 105.0000000000, 225.0000000000, 104.0000000000, 105.0000000000, 116.↩
0000000000, 117.0000000000, 106.0000000000, 106.0000000000, 107.0000000000, 101.0000000000,
118.0000000000, 117.0000000000, 123.0000000000, 118.0000000000, 108.0000000000, 107.↩
0000000000, 114.0000000000, 102.0000000000, 119.0000000000, 113.0000000000, 120.0000000000,
108.0000000000, 109.0000000000, 119.0000000000, 103.0000000000, 120.0000000000, 121.↩
0000000000, 61.0000000000, 54.0000000000, 109.0000000000, 117.0000000000, 122.0000000000,
110.0000000000, 225.0000000000, 230.0000000000, 111.0000000000, 105.0000000000, 116.↩
0000000000, 123.0000000000, 122.0000000000, 111.0000000000, 112.0000000000, 106.0000000000,
123.0000000000, 107.0000000000, 112.0000000000, 119.0000000000, 124.0000000000, 113.↩
0000000000, 114.0000000000, 130.0000000000, 125.0000000000, 124.0000000000, 113.0000000000,
120.0000000000, 118.0000000000, 131.0000000000, 115.0000000000, 114.0000000000, 126.↩

0000000000, 125.0000000000, 121.0000000000, 109.0000000000, 119.0000000000, 61.0000000000, 132.0000000000, 115.0000000000, 126.0000000000, 120.0000000000, 127.0000000000, 68.0000000000, 123.0000000000, 116.0000000000, 230.0000000000, 235.0000000000, 117.0000000000, 128.↩ 0000000000, 122.0000000000, 117.0000000000, 112.0000000000, 124.0000000000, 118.0000000000, 133.0000000000, 128.0000000000, 129.0000000000, 129.0000000000, 118.0000000000, 123.↩ 0000000000, 130.0000000000, 119.0000000000, 130.0000000000, 120.0000000000, 119.0000000000, 131.0000000000, 131.0000000000, 121.0000000000, 120.0000000000, 132.0000000000, 73.0000000000, 68.0000000000, 121.0000000000, 133.0000000000, 235.0000000000, 240.0000000000, 122.0000000000, 123.0000000000, 130.0000000000, 123.0000000000, 133.0000000000, 137.0000000000, 124.↩ 0000000000, 134.0000000000, 124.0000000000, 134.0000000000, 135.0000000000, 125.0000000000, 119.0000000000, 129.0000000000, 131.0000000000, 136.0000000000, 132.0000000000, 139.↩ 0000000000, 135.0000000000, 130.0000000000, 125.0000000000, 120.0000000000, 126.0000000000, 126.0000000000, 121.0000000000, 131.0000000000, 127.0000000000, 240.0000000000, 128.↩ 0000000000, 137.0000000000, 123.0000000000, 129.0000000000, 137.0000000000, 129.0000000000, 130.0000000000, 138.0000000000, 135.0000000000, 130.0000000000, 141.0000000000, 138.↩ 0000000000, 134.0000000000, 139.0000000000, 131.0000000000, 131.0000000000, 132.0000000000, 138.0000000000, 140.0000000000, 240.0000000000, 129.0000000000, 133.0000000000, 134.↩ 0000000000, 250.0000000000, 245.0000000000, 137.0000000000, 141.0000000000, 135.0000000000, 134.0000000000, 140.0000000000, 136.0000000000, 131.0000000000, 135.0000000000, 255.↩ 0000000000, 138.0000000000, 141.0000000000, 142.0000000000, 137.0000000000, 250.0000000000, 139.0000000000, 140.0000000000, 135.0000000000, 138.0000000000, 140.0000000000, 141.↩ 0000000000, 21.0000000000, 144.0000000000, 20.0000000000, 146.0000000000, 143.0000000000, 28.↩ 0000000000, 21.0000000000, 146.0000000000, 147.0000000000, 146.0000000000, 149.0000000000, 143.0000000000, 149.0000000000, 145.0000000000, 144.0000000000, 150.0000000000, 147.↩ 0000000000, 143.0000000000, 150.0000000000, 28.0000000000, 151.0000000000, 146.0000000000, 144.0000000000, 149.0000000000, 145.0000000000, 145.0000000000, 152.0000000000, 148.↩ 0000000000, 153.0000000000, 154.0000000000, 146.0000000000, 150.0000000000, 154.0000000000, 155.0000000000, 147.0000000000, 149.0000000000, 151.0000000000, 160.0000000000, 146.↩ 0000000000, 156.0000000000, 42.0000000000, 35.0000000000, 155.0000000000, 28.0000000000, 147.↩ 0000000000, 150.0000000000, 148.0000000000, 149.0000000000, 153.0000000000, 157.0000000000, 158.0000000000, 159.0000000000, 154.0000000000, 152.0000000000, 149.0000000000, 149.↩ 0000000000, 159.0000000000, 153.0000000000, 150.0000000000, 166.0000000000, 160.0000000000, 160.0000000000, 161.0000000000, 156.0000000000, 151.0000000000, 150.0000000000, 155.↩ 0000000000, 162.0000000000, 161.0000000000, 151.0000000000, 168.0000000000, 42.0000000000, 164.0000000000, 153.0000000000, 158.0000000000, 152.0000000000, 157.0000000000, 159.↩ 0000000000, 165.0000000000, 153.0000000000, 164.0000000000, 154.0000000000, 158.0000000000, 165.0000000000, 153.0000000000, 171.0000000000, 166.0000000000, 167.0000000000, 154.↩ 0000000000, 155.0000000000, 150.0000000000, 166.0000000000, 161.0000000000, 155.0000000000, 168.0000000000, 167.0000000000, 156.0000000000, 173.0000000000, 160.0000000000, 49.0000000000, 42.0000000000, 168.0000000000, 56.0000000000, 156.0000000000, 169.0000000000, 164.0000000000, 157.0000000000, 165.0000000000, 163.0000000000, 214.0000000000, 215.0000000000, 170.↩ 0000000000, 157.0000000000, 158.0000000000, 164.0000000000, 171.0000000000, 170.0000000000, 159.0000000000, 176.0000000000, 158.0000000000, 172.0000000000, 159.0000000000, 177.↩ 0000000000, 171.0000000000, 154.0000000000, 167.0000000000, 160.0000000000, 173.0000000000, 172.0000000000, 178.0000000000, 166.0000000000, 160.0000000000, 161.0000000000, 162.↩ 0000000000, 173.0000000000, 156.0000000000, 174.0000000000, 161.0000000000, 179.0000000000, 169.0000000000, 63.0000000000, 56.0000000000, 174.0000000000, 175.0000000000, 168.0000000000, 162.0000000000, 176.0000000000, 215.0000000000, 222.0000000000, 164.0000000000, 227.↩ 0000000000, 165.0000000000, 159.0000000000, 165.0000000000, 182.0000000000, 176.0000000000, 166.0000000000, 177.0000000000, 183.0000000000, 178.0000000000, 177.0000000000, 166.↩ 0000000000, 167.0000000000, 178.0000000000, 179.0000000000, 167.0000000000, 168.0000000000, 161.0000000000, 184.0000000000, 179.0000000000, 180.0000000000, 168.0000000000, 175.↩ 0000000000, 169.0000000000, 185.0000000000, 180.0000000000, 186.0000000000, 174.0000000000, 169.0000000000, 181.0000000000, 70.0000000000, 63.0000000000, 227.0000000000, 165.0000000000, 182.0000000000, 170.0000000000, 171.0000000000, 171.0000000000, 183.0000000000, 182.↩ 0000000000, 166.0000000000, 172.0000000000, 188.0000000000, 183.0000000000, 184.0000000000, 172.0000000000, 167.0000000000, 173.0000000000, 168.0000000000, 190.0000000000, 174.↩ 0000000000, 184.0000000000, 173.0000000000, 185.0000000000, 175.0000000000, 186.0000000000,

174.0000000000, 185.0000000000, 175.0000000000, 192.0000000000, 186.0000000000, 70.0000000000, 187.0000000000, 171.0000000000, 176.0000000000, 227.0000000000, 237.0000000000, 232.↩ 0000000000, 177.0000000000, 188.0000000000, 189.0000000000, 184.0000000000, 188.0000000000, 172.0000000000, 178.0000000000, 177.0000000000, 189.0000000000, 178.0000000000, 179.↩ 0000000000, 190.0000000000, 183.0000000000, 173.0000000000, 174.0000000000, 190.0000000000, 179.0000000000, 186.0000000000, 180.0000000000, 191.0000000000, 191.0000000000, 175.↩ 0000000000, 192.0000000000, 185.0000000000, 180.0000000000, 181.0000000000, 193.0000000000, 70.0000000000, 74.0000000000, 192.0000000000, 181.0000000000, 189.0000000000, 194.0000000000, 182.0000000000, 237.0000000000, 177.0000000000, 183.0000000000, 242.0000000000, 188.↩ 0000000000, 190.0000000000, 195.0000000000, 184.0000000000, 183.0000000000, 194.0000000000, 189.0000000000, 185.0000000000, 200.0000000000, 196.0000000000, 179.0000000000, 195.↩ 0000000000, 184.0000000000, 191.0000000000, 196.0000000000, 192.0000000000, 201.0000000000, 197.0000000000, 185.0000000000, 186.0000000000, 190.0000000000, 186.0000000000, 181.↩ 0000000000, 193.0000000000, 198.0000000000, 197.0000000000, 191.0000000000, 187.0000000000, 198.0000000000, 187.0000000000, 74.0000000000, 192.0000000000, 199.0000000000, 189.0000000000, 188.0000000000, 242.0000000000, 195.0000000000, 199.0000000000, 200.0000000000, 203.↩ 0000000000, 190.0000000000, 194.0000000000, 189.0000000000, 191.0000000000, 190.0000000000, 201.0000000000, 200.0000000000, 198.0000000000, 192.0000000000, 205.0000000000, 201.↩ 0000000000, 191.0000000000, 202.0000000000, 202.0000000000, 192.0000000000, 197.0000000000, 242.0000000000, 247.0000000000, 203.0000000000, 252.0000000000, 195.0000000000, 194.↩ 0000000000, 195.0000000000, 190.0000000000, 203.0000000000, 204.0000000000, 201.0000000000, 196.0000000000, 207.0000000000, 196.0000000000, 197.0000000000, 191.0000000000, 200.↩ 0000000000, 204.0000000000, 205.0000000000, 197.0000000000, 205.0000000000, 206.0000000000, 204.0000000000, 195.0000000000, 200.0000000000, 252.0000000000, 257.0000000000, 199.↩ 0000000000, 203.0000000000, 201.0000000000, 200.0000000000, 207.0000000000, 206.0000000000, 207.0000000000, 197.0000000000, 201.0000000000, 204.0000000000, 257.0000000000, 203.↩ 0000000000, 207.0000000000, 208.0000000000, 261.0000000000, 206.0000000000, 201.0000000000, 208.0000000000, 204.0000000000, 206.0000000000, 261.0000000000, 211.0000000000, 97.0000000000, 210.0000000000, 211.0000000000, 97.0000000000, 212.0000000000, 213.0000000000, 218.0000000000, 209.0000000000, 216.0000000000, 210.0000000000, 217.0000000000, 212.0000000000, 223.↩ 0000000000, 210.0000000000, 217.0000000000, 211.0000000000, 218.0000000000, 224.0000000000, 104.0000000000, 218.0000000000, 210.0000000000, 219.0000000000, 97.0000000000, 163.↩ 0000000000, 164.0000000000, 221.0000000000, 215.0000000000, 164.0000000000, 222.0000000000, 214.0000000000, 221.0000000000, 170.0000000000, 217.0000000000, 220.0000000000, 223.↩ 0000000000, 221.0000000000, 212.0000000000, 220.0000000000, 216.0000000000, 211.0000000000, 223.0000000000, 224.0000000000, 213.0000000000, 212.0000000000, 210.0000000000, 228.↩ 0000000000, 224.0000000000, 225.0000000000, 104.0000000000, 110.0000000000, 229.0000000000, 213.0000000000, 214.0000000000, 221.0000000000, 217.0000000000, 214.0000000000, 223.↩ 0000000000, 215.0000000000, 226.0000000000, 217.0000000000, 220.0000000000, 222.0000000000, 170.0000000000, 215.0000000000, 226.0000000000, 227.0000000000, 221.0000000000, 221.↩ 0000000000, 228.0000000000, 217.0000000000, 218.0000000000, 212.0000000000, 226.0000000000, 233.0000000000, 228.0000000000, 213.0000000000, 229.0000000000, 219.0000000000, 218.↩ 0000000000, 219.0000000000, 229.0000000000, 110.0000000000, 116.0000000000, 230.0000000000, 234.0000000000, 223.0000000000, 228.0000000000, 222.0000000000, 221.0000000000, 231.↩ 0000000000, 227.0000000000, 231.0000000000, 232.0000000000, 176.0000000000, 226.0000000000, 222.0000000000, 170.0000000000, 182.0000000000, 224.0000000000, 233.0000000000, 231.↩ 0000000000, 226.0000000000, 218.0000000000, 223.0000000000, 236.0000000000, 233.0000000000, 234.0000000000, 219.0000000000, 225.0000000000, 224.0000000000, 235.0000000000, 122.↩ 0000000000, 225.0000000000, 234.0000000000, 116.0000000000, 226.0000000000, 236.0000000000, 228.0000000000, 232.0000000000, 227.0000000000, 231.0000000000, 227.0000000000, 182.↩ 0000000000, 236.0000000000, 237.0000000000, 234.0000000000, 236.0000000000, 224.0000000000, 238.0000000000, 229.0000000000, 241.0000000000, 228.0000000000, 225.0000000000, 238.↩ 0000000000, 230.0000000000, 243.0000000000, 229.0000000000, 235.0000000000, 233.0000000000, 239.0000000000, 239.0000000000, 240.0000000000, 128.0000000000, 122.0000000000, 234.↩ 0000000000, 230.0000000000, 231.0000000000, 228.0000000000, 232.0000000000, 237.0000000000, 241.0000000000, 233.0000000000, 182.0000000000, 242.0000000000, 236.0000000000, 232.↩ 0000000000, 188.0000000000, 241.0000000000, 243.0000000000, 246.0000000000, 233.0000000000, 234.0000000000, 241.0000000000, 244.0000000000, 248.0000000000, 243.0000000000, 234.↩

0000000000, 235.0000000000, 240.0000000000, 137.0000000000, 249.0000000000, 128.0000000000, 235.0000000000, 133.0000000000, 239.0000000000, 245.0000000000, 244.0000000000, 242.↩
0000000000, 233.0000000000, 236.0000000000, 237.0000000000, 238.0000000000, 246.0000000000, 241.0000000000, 194.0000000000, 199.0000000000, 247.0000000000, 237.0000000000, 251.↩
0000000000, 188.0000000000, 246.0000000000, 234.0000000000, 248.0000000000, 239.0000000000, 251.0000000000, 238.0000000000, 246.0000000000, 248.0000000000, 249.0000000000, 253.↩
0000000000, 240.0000000000, 239.0000000000, 249.0000000000, 137.0000000000, 254.0000000000, 240.0000000000, 250.0000000000, 241.0000000000, 238.0000000000, 251.0000000000, 242.↩
0000000000, 243.0000000000, 251.0000000000, 199.0000000000, 252.0000000000, 242.0000000000, 251.0000000000, 243.0000000000, 239.0000000000, 244.0000000000, 253.0000000000, 244.↩
0000000000, 254.0000000000, 258.0000000000, 253.0000000000, 240.0000000000, 245.0000000000, 137.0000000000, 140.0000000000, 254.0000000000, 245.0000000000, 255.0000000000, 259.↩
0000000000, 253.0000000000, 243.0000000000, 252.0000000000, 256.0000000000, 242.0000000000, 247.0000000000, 248.0000000000, 246.0000000000, 251.0000000000, 203.0000000000, 199.↩
0000000000, 247.0000000000, 256.0000000000, 257.0000000000, 260.0000000000, 260.0000000000, 256.0000000000, 258.0000000000, 244.0000000000, 248.0000000000, 249.0000000000, 251.↩
0000000000, 258.0000000000, 245.0000000000, 259.0000000000, 249.0000000000, 250.0000000000, 140.0000000000, 142.0000000000, 250.0000000000, 260.0000000000, 253.0000000000, 251.↩
0000000000, 252.0000000000, 252.0000000000, 203.0000000000, 263.0000000000, 261.0000000000, 206.0000000000, 260.0000000000, 263.0000000000, 262.0000000000, 249.0000000000, 260.↩
0000000000, 254.0000000000, 253.0000000000, 259.0000000000, 250.0000000000, 255.0000000000, 258.0000000000, 254.0000000000, 256.0000000000, 252.0000000000, 253.0000000000, 258.↩
0000000000, 263.0000000000, 257.0000000000, 263.0000000000, 206.0000000000, 257.0000000000, 258.0000000000, 259.0000000000, 262.0000000000, 260.0000000000, 258.0000000000, 257.↩
0000000000],[0.0000000000, 1.0000000000, 2.0000000000, 3.0000000000, 4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000, 9.0000000000, 10.0000000000, 11.0000000000, 12.↩
0000000000, 13.0000000000, 14.0000000000, 15.0000000000, 16.0000000000, 17.0000000000, 18.↩
0000000000, 19.0000000000, 20.0000000000, 21.0000000000, 22.0000000000, 23.0000000000, 24.↩
0000000000, 25.0000000000, 26.0000000000, 27.0000000000, 28.0000000000, 29.0000000000, 30.↩
0000000000, 31.0000000000, 32.0000000000, 33.0000000000, 34.0000000000, 35.0000000000, 36.↩
0000000000, 37.0000000000, 38.0000000000, 39.0000000000, 40.0000000000, 41.0000000000, 42.↩
0000000000, 43.0000000000, 44.0000000000, 45.0000000000, 46.0000000000, 47.0000000000, 48.↩
0000000000, 49.0000000000, 50.0000000000, 51.0000000000, 52.0000000000, 53.0000000000, 54.↩
0000000000, 55.0000000000, 56.0000000000, 57.0000000000, 58.0000000000, 59.0000000000, 60.↩
0000000000, 61.0000000000, 62.0000000000, 63.0000000000, 64.0000000000, 65.0000000000, 66.↩
0000000000, 67.0000000000, 68.0000000000, 69.0000000000, 70.0000000000, 71.0000000000, 72.↩
0000000000, 73.0000000000, 74.0000000000, 75.0000000000, 76.0000000000, 77.0000000000, 78.↩
0000000000, 79.0000000000, 80.0000000000, 81.0000000000, 82.0000000000, 83.0000000000, 84.↩
0000000000, 85.0000000000, 86.0000000000, 87.0000000000, 88.0000000000, 89.0000000000, 90.↩
0000000000, 91.0000000000, 92.0000000000, 93.0000000000, 94.0000000000, 95.0000000000, 96.↩
0000000000, 97.0000000000, 98.0000000000, 99.0000000000, 100.0000000000, 101.0000000000, 102.↩
0000000000, 103.0000000000, 104.0000000000, 105.0000000000, 106.0000000000, 107.0000000000, 108.0000000000, 109.0000000000, 110.0000000000, 111.0000000000, 112.0000000000, 113.↩
0000000000, 114.0000000000, 115.0000000000, 116.0000000000, 117.0000000000, 118.0000000000, 119.0000000000, 120.0000000000, 121.0000000000, 122.0000000000, 123.0000000000, 124.↩
0000000000, 125.0000000000, 126.0000000000, 127.0000000000, 128.0000000000, 129.0000000000, 130.0000000000, 131.0000000000, 132.0000000000, 133.0000000000, 134.0000000000, 135.↩
0000000000, 136.0000000000, 137.0000000000, 138.0000000000, 139.0000000000, 140.0000000000, 141.0000000000, 142.0000000000, 143.0000000000, 144.0000000000, 145.0000000000, 146.↩
0000000000, 147.0000000000, 148.0000000000, 149.0000000000, 150.0000000000, 151.0000000000, 152.0000000000, 153.0000000000, 154.0000000000, 155.0000000000, 156.0000000000, 157.↩
0000000000, 158.0000000000, 159.0000000000, 160.0000000000, 161.0000000000, 162.0000000000, 163.0000000000, 164.0000000000, 165.0000000000, 166.0000000000, 167.0000000000, 168.↩
0000000000, 169.0000000000, 170.0000000000, 171.0000000000, 172.0000000000, 173.0000000000, 174.0000000000, 175.0000000000, 176.0000000000, 177.0000000000, 178.0000000000, 179.↩
0000000000, 180.0000000000, 181.0000000000, 182.0000000000, 183.0000000000, 184.0000000000, 185.0000000000, 186.0000000000, 187.0000000000, 188.0000000000, 189.0000000000, 190.↩
0000000000, 191.0000000000, 192.0000000000, 193.0000000000, 194.0000000000, 195.0000000000,

196.0000000000, 197.0000000000, 198.0000000000, 199.0000000000, 200.0000000000, 201.↩
0000000000, 202.0000000000, 203.0000000000, 204.0000000000, 205.0000000000, 206.0000000000,
207.0000000000, 208.0000000000, 209.0000000000, 210.0000000000, 211.0000000000, 212.↩
0000000000, 213.0000000000, 214.0000000000, 215.0000000000, 216.0000000000, 217.0000000000,
218.0000000000, 219.0000000000, 220.0000000000, 221.0000000000, 222.0000000000, 223.↩
0000000000, 224.0000000000, 225.0000000000, 226.0000000000, 227.0000000000, 228.0000000000,
229.0000000000, 230.0000000000, 231.0000000000, 232.0000000000, 233.0000000000, 234.↩
0000000000, 235.0000000000, 236.0000000000, 237.0000000000, 238.0000000000, 239.0000000000,
240.0000000000, 241.0000000000, 242.0000000000, 243.0000000000, 244.0000000000, 245.↩
0000000000, 246.0000000000, 247.0000000000, 248.0000000000, 249.0000000000, 250.0000000000,
251.0000000000, 252.0000000000, 253.0000000000, 254.0000000000, 255.0000000000, 256.↩
0000000000, 257.0000000000, 258.0000000000, 259.0000000000, 260.0000000000, 261.0000000000,
262.0000000000, 263.0000000000, 0.0000000000, 0.0000000000, 1.0000000000, 1.0000000000, 1.↩
0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 2.0000000000, 2.0000000000,
2.0000000000, 2.0000000000, 3.0000000000, 3.0000000000, 3.0000000000, 3.0000000000, 3.↩
0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000,
5.0000000000, 5.0000000000, 5.0000000000, 6.0000000000, 6.0000000000, 6.0000000000, 6.↩
0000000000, 6.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 8.0000000000,
8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 9.↩
0000000000, 9.0000000000, 9.0000000000, 9.0000000000, 9.0000000000, 10.0000000000, 10.↩
0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 11.↩
0000000000, 11.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.↩
0000000000, 12.0000000000, 13.0000000000, 13.0000000000, 13.0000000000, 14.0000000000, 14.↩
0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 15.0000000000, 15.↩
0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 16.0000000000, 16.↩
0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 17.0000000000, 17.↩
0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 18.↩
0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.↩
0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.↩
0000000000, 20.0000000000, 20.0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 21.↩
0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 22.0000000000, 22.0000000000, 22.↩
0000000000, 22.0000000000, 22.0000000000, 22.0000000000, 23.0000000000, 23.0000000000, 23.↩
0000000000, 23.0000000000, 23.0000000000, 24.0000000000, 24.0000000000, 24.0000000000, 24.↩
0000000000, 24.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.↩
0000000000, 25.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 26.↩
0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.↩
0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.↩
0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.↩
0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.↩
0000000000, 30.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.↩
0000000000, 31.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.↩
0000000000, 32.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.↩
0000000000, 33.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.↩
0000000000, 34.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.↩
0000000000, 35.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 36.↩
0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.↩
0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.↩
0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.↩
0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.↩
0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.↩
0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.↩
0000000000, 42.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.↩
0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 44.0000000000, 44.0000000000, 44.↩
0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 45.0000000000, 45.↩
0000000000, 45.0000000000, 45.0000000000, 45.0000000000, 46.0000000000, 46.0000000000, 46.↩
0000000000, 46.0000000000, 46.0000000000, 46.0000000000, 47.0000000000, 47.0000000000, 47.↩
0000000000, 47.0000000000, 47.0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 48.↩
0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 49.0000000000, 49.0000000000, 49.↩

0000000000, 49.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 50.↩
0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 52.↩
0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.↩
0000000000, 52.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 53.↩
0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.↩
0000000000, 54.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 55.↩
0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.↩
0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.↩
0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 59.↩
0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 60.↩
0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 61.↩
0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.↩
0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.↩
0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.↩
0000000000, 63.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.↩
0000000000, 64.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.↩
0000000000, 65.0000000000, 65.0000000000, 66.0000000000, 66.0000000000, 67.0000000000, 67.↩
0000000000, 67.0000000000, 67.0000000000, 68.0000000000, 68.0000000000, 68.0000000000, 68.↩
0000000000, 68.0000000000, 68.0000000000, 69.0000000000, 69.0000000000, 69.0000000000, 70.↩
0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.↩
0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 72.↩
0000000000, 72.0000000000, 72.0000000000, 73.0000000000, 73.0000000000, 74.0000000000, 74.↩
0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 75.0000000000, 75.↩
0000000000, 75.0000000000, 75.0000000000, 76.0000000000, 76.0000000000, 77.0000000000, 77.↩
0000000000, 77.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.↩
0000000000, 78.0000000000, 79.0000000000, 79.0000000000, 79.0000000000, 79.0000000000, 80.↩
0000000000, 80.0000000000, 80.0000000000, 80.0000000000, 81.0000000000, 81.0000000000, 81.↩
0000000000, 81.0000000000, 81.0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 82.↩
0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 83.0000000000, 83.0000000000, 84.↩
0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.↩
0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.↩
0000000000, 86.0000000000, 86.0000000000, 86.0000000000, 86.0000000000, 87.0000000000, 87.↩
0000000000, 87.0000000000, 87.0000000000, 87.0000000000, 88.0000000000, 88.0000000000, 88.↩
0000000000, 88.0000000000, 88.0000000000, 88.0000000000, 89.0000000000, 89.0000000000, 89.↩
0000000000, 89.0000000000, 89.0000000000, 89.0000000000, 90.0000000000, 90.0000000000, 90.↩
0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 91.↩
0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 92.↩
0000000000, 92.0000000000, 92.0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 93.↩
0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 94.0000000000, 94.0000000000, 94.↩
0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 95.0000000000, 95.↩
0000000000, 95.0000000000, 95.0000000000, 95.0000000000, 96.0000000000, 96.0000000000, 96.↩
0000000000, 96.0000000000, 96.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.↩
0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 98.0000000000, 98.↩
0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 100.↩
0000000000, 100.0000000000, 100.0000000000, 100.0000000000, 101.0000000000, 101.0000000000,
101.0000000000, 101.0000000000, 101.0000000000, 101.0000000000, 102.0000000000, 102.↩
0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000,
103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.↩
0000000000, 103.0000000000, 104.0000000000, 104.0000000000, 104.0000000000, 104.0000000000,
104.0000000000, 104.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.↩
0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 106.0000000000,
106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.↩
0000000000, 107.0000000000, 107.0000000000, 107.0000000000, 107.0000000000, 107.0000000000,
108.0000000000, 108.0000000000, 108.0000000000, 108.0000000000, 108.0000000000, 109.↩
0000000000, 109.0000000000, 109.0000000000, 109.0000000000, 109.0000000000, 110.0000000000,
110.0000000000, 110.0000000000, 110.0000000000, 110.0000000000, 111.0000000000, 111.↩
0000000000, 111.0000000000, 111.0000000000, 112.0000000000, 112.0000000000, 112.0000000000,
112.0000000000, 112.0000000000, 112.0000000000, 113.0000000000, 113.0000000000, 113.↩

0000000000, 113.0000000000, 113.0000000000, 113.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 115.0000000000, 115.↩
0000000000, 115.0000000000, 115.0000000000, 115.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 117.↩
0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.↩
0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.↩
0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.↩
0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.↩
0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 124.0000000000, 124.0000000000, 124.0000000000, 124.0000000000, 124.0000000000, 125.0000000000, 125.0000000000, 125.↩
0000000000, 125.0000000000, 126.0000000000, 126.0000000000, 126.0000000000, 126.0000000000, 127.0000000000, 127.0000000000, 127.0000000000, 128.0000000000, 128.0000000000, 128.↩
0000000000, 128.0000000000, 128.0000000000, 129.0000000000, 129.0000000000, 129.0000000000, 129.0000000000, 129.0000000000, 129.0000000000, 130.0000000000, 130.0000000000, 130.↩
0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.↩
0000000000, 131.0000000000, 132.0000000000, 132.0000000000, 132.0000000000, 132.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 134.↩
0000000000, 134.0000000000, 134.0000000000, 134.0000000000, 134.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 136.↩
0000000000, 136.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 138.0000000000, 138.↩
0000000000, 138.0000000000, 138.0000000000, 138.0000000000, 139.0000000000, 139.0000000000, 139.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 140.↩
0000000000, 140.0000000000, 141.0000000000, 141.0000000000, 141.0000000000, 141.0000000000, 142.0000000000, 142.0000000000, 143.0000000000, 143.0000000000, 143.0000000000, 143.↩
0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 145.0000000000, 145.0000000000, 145.0000000000, 146.0000000000, 146.0000000000, 146.↩
0000000000, 146.0000000000, 146.0000000000, 146.0000000000, 147.0000000000, 147.0000000000, 147.0000000000, 147.0000000000, 147.0000000000, 148.0000000000, 148.0000000000, 149.↩
0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.↩
0000000000, 150.0000000000, 150.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 152.0000000000, 152.↩
0000000000, 152.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 154.0000000000, 154.0000000000, 154.↩
0000000000, 154.0000000000, 154.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 156.0000000000, 156.0000000000, 156.↩
0000000000, 156.0000000000, 156.0000000000, 157.0000000000, 157.0000000000, 157.0000000000, 157.0000000000, 158.0000000000, 158.0000000000, 158.0000000000, 158.0000000000, 158.↩
0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 160.↩
0000000000, 160.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 162.0000000000, 162.0000000000, 162.0000000000, 162.↩
0000000000, 162.0000000000, 162.0000000000, 163.0000000000, 163.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.↩
0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.↩
0000000000, 166.0000000000, 166.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 168.0000000000, 168.0000000000, 168.↩
0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 170.↩
0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.↩

0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 172.0000000000,
173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.↩
0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000,
174.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 175.↩
0000000000, 175.0000000000, 175.0000000000, 176.0000000000, 176.0000000000, 176.0000000000,
176.0000000000, 176.0000000000, 177.0000000000, 177.0000000000, 177.0000000000, 177.↩
0000000000, 177.0000000000, 177.0000000000, 178.0000000000, 178.0000000000, 178.0000000000,
178.0000000000, 178.0000000000, 179.0000000000, 179.0000000000, 179.0000000000, 179.↩
0000000000, 179.0000000000, 179.0000000000, 180.0000000000, 180.0000000000, 180.0000000000,
180.0000000000, 181.0000000000, 181.0000000000, 181.0000000000, 181.0000000000, 181.↩
0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000,
182.0000000000, 182.0000000000, 183.0000000000, 183.0000000000, 183.0000000000, 183.↩
0000000000, 183.0000000000, 183.0000000000, 184.0000000000, 184.0000000000, 184.0000000000,
184.0000000000, 184.0000000000, 184.0000000000, 185.0000000000, 185.0000000000, 185.↩
0000000000, 185.0000000000, 185.0000000000, 185.0000000000, 186.0000000000, 186.0000000000,
186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 187.0000000000, 187.↩
0000000000, 187.0000000000, 187.0000000000, 187.0000000000, 188.0000000000, 188.0000000000,
188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 189.↩
0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000,
190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.↩
0000000000, 190.0000000000, 190.0000000000, 191.0000000000, 191.0000000000, 191.0000000000,
191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 192.0000000000, 192.↩
0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000,
193.0000000000, 193.0000000000, 193.0000000000, 193.0000000000, 194.0000000000, 194.↩
0000000000, 194.0000000000, 194.0000000000, 194.0000000000, 195.0000000000, 195.0000000000,
195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 196.0000000000, 196.↩
0000000000, 196.0000000000, 196.0000000000, 197.0000000000, 197.0000000000, 197.0000000000,
197.0000000000, 197.0000000000, 197.0000000000, 198.0000000000, 198.0000000000, 198.↩
0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000,
199.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 200.↩
0000000000, 200.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000,
201.0000000000, 201.0000000000, 201.0000000000, 202.0000000000, 202.0000000000, 203.↩
0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000,
203.0000000000, 204.0000000000, 204.0000000000, 204.0000000000, 204.0000000000, 204.↩
0000000000, 205.0000000000, 205.0000000000, 205.0000000000, 206.0000000000, 206.0000000000,
206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 207.0000000000, 207.↩
0000000000, 207.0000000000, 207.0000000000, 208.0000000000, 208.0000000000, 209.0000000000,
209.0000000000, 209.0000000000, 210.0000000000, 210.0000000000, 210.0000000000, 210.↩
0000000000, 210.0000000000, 210.0000000000, 211.0000000000, 211.0000000000, 211.0000000000,
211.0000000000, 212.0000000000, 212.0000000000, 212.0000000000, 212.0000000000, 212.↩
0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000,
213.0000000000, 214.0000000000, 214.0000000000, 214.0000000000, 214.0000000000, 215.↩
0000000000, 215.0000000000, 215.0000000000, 215.0000000000, 215.0000000000, 216.0000000000,
216.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 217.↩
0000000000, 217.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000,
218.0000000000, 218.0000000000, 219.0000000000, 219.0000000000, 219.0000000000, 219.↩
0000000000, 219.0000000000, 219.0000000000, 220.0000000000, 220.0000000000, 220.0000000000,
221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.↩
0000000000, 221.0000000000, 222.0000000000, 222.0000000000, 222.0000000000, 222.0000000000,
222.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 223.↩
0000000000, 223.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000,
224.0000000000, 224.0000000000, 225.0000000000, 225.0000000000, 225.0000000000, 225.↩
0000000000, 225.0000000000, 225.0000000000, 226.0000000000, 226.0000000000, 226.0000000000,
226.0000000000, 226.0000000000, 226.0000000000, 227.0000000000, 227.0000000000, 227.↩
0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 228.0000000000,
228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.↩
0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 229.0000000000,
230.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 231.↩

0000000000, 231.0000000000, 231.0000000000, 231.0000000000, 231.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 233.0000000000, 233.↩
0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.↩
0000000000, 234.0000000000, 234.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 236.0000000000, 236.0000000000, 236.↩
0000000000, 236.0000000000, 236.0000000000, 236.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 238.0000000000, 238.↩
0000000000, 238.0000000000, 238.0000000000, 238.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 240.0000000000, 240.↩
0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 241.0000000000, 241.0000000000, 241.0000000000, 241.0000000000, 241.↩
0000000000, 241.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 243.0000000000, 243.↩
0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 245.0000000000, 245.↩
0000000000, 245.0000000000, 245.0000000000, 245.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 247.0000000000, 247.0000000000, 247.↩
0000000000, 247.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 249.↩
0000000000, 249.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 251.0000000000, 251.0000000000, 251.↩
0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.↩
0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 254.0000000000, 254.0000000000, 254.↩
0000000000, 254.0000000000, 255.0000000000, 255.0000000000, 255.0000000000, 256.0000000000, 256.0000000000, 256.0000000000, 256.0000000000, 257.0000000000, 257.0000000000, 257.↩
0000000000, 257.0000000000, 257.0000000000, 257.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 259.↩
0000000000, 259.0000000000, 259.0000000000, 259.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 261.0000000000, 261.↩
0000000000, 261.0000000000, 262.0000000000, 262.0000000000, 263.0000000000, 263.0000000000, 263.0000000000, 263.0000000000, 263.0000000000]

- list calcWeightsGaussCorrect = [[-0.57735026918962573105886 80, 0.57735026918962573105886 80],[1.↩
0000000000000004440892099, 1.0000000000000004440892099]]
- list gaußLegendreCorrect = [[0.21132486540518713, 0.21132486540518713, 0.7886751345948129, 0.7886751345948129], [0.16666666666666669, 0.6220084679281462, 0.044658198738520456, 0.↩
16666666666666669], [0.19716878364870338, 0.19716878364870338, 0.052831216351296825, 0.↩
052831216351296825]]
- list reducedSFCorrect = [-0.27937980331459932870075136, -0.33966732733873783445233130, 158.↩
30535989173870348167838529, 111.95922396290268352458951995, 58.10903859010657157568857656, 193.39790162602255918500304688, 268.25033740186444219943950884, 189.82215394456304125014867168, 0.25883372957144956671982072, 98.74870633402809971812530421, 215.53653410699894266144838184, 327.39158447388479089568136260, 284.96744613445838467669091187, 202.31101004349304162133194041, 0.82513685120877511280923500, 105.22484885820357192187657347, 223.08256339289710012963041663, 364.76793579171919645887101069, 347.61723438903078431394533254, 186.13386039326010745753592346, 132.63331666423582078095932957, 0.93045442511324694123686641, 69.59377646537200234888587147, 377.49981102452846926098573022, 387.86131916037487599169253372, 228.06934028761415333974582609, -0.39282069172599420880942489, 0.28340911989108974466944346, 1.32922047073893168089853134, 0.77036064878831922442259383, 402.01980094098689733073115349, 253.64580857732585172925610095, -0.71137220262248490598722128, -187.02281867595976905249699485, -132.12396908339931655973487068, 1.00935338967148879874002887, -68.31938968731270733769633807, 262.84597566425361492292722687, -1.01378116441184573659484158, -229.49291759567395843077974860, -285.00916079306904293844127096, -202.15395986486728929776290897, 0.20369749822588545384860481, -105.12807714412967641237628413, -1.65128343645850605980740511, -255.65451390973146317264763638, -348.80806087323372821629163809, -268.84160316836425863584736362, -190.14226619109740568092092872, -0.35372141836276682624884415, -98.67697413471481127089646179, -265.72733211685635978938080370, -390.53074044891235416798735969,

-328.54385597348704095566063188, -159.10063256399396891538344789, -112.66830824900223717577318894, -0.48136222438574449711268244, -58.492965775446009502047672870, -406.26733909792579879649565555, -367.00125201198864033358404413, -194.68519835904922388181148563, -0.38739421188725309264100360, -380.72973951162691719218855724, -216.95201141250419141215388663, -224.64681969190800714386568870, -57.64885629013803480802380363, -111.76778906363414023417135468, -97.23151838524054824119957630, -158.57787734712869109898747411, -189.13083157351150020986096933, -103.47360646666479055966192391, -194.37812613590944010866223834, -268.14533020187656120469910093, -201.00144759920507908645959105, -67.33160591182809184829238802, -216.97908089605186887638410553, -328.52149241710583282838342711, -284.82532200213120177068049088, -130.61038743255170402335352264, 1.17408313059079461027067737, -225.01886593214194931533711497, -366.28282561516010673585697077, -347.97810273027931771139265038, -184.84955361689296182703401428, 1.3557588455271485372577447, 68.838658634642342803999781 61, -379.59834765834438030651654117, -389.38131238244437781759188510, -226.65542721376317558679147623, 2.45142690491787362816467066, 132.61870525252811603422742337, 104.51756439557146904917317443, -402.65102785189810674637556076, -252.04129957093991265537624713, 2.66640808738427992352626461, 188.46031263508777442439168226, 202.68330394428298291131795850, 98.04919160612566031431924785, -262.11224472208607494394527748, 2.80043348997752872264754842, 231.30365585971119912755966652, 286.80511684477886547028901987, 190.23232474250315249264531303, 57.84805143771897917304158909, 2.85708460601647828980276245, 258.69203442119561486833845265, 350.58867061999404768357635476, 268.52259877181154479330871254, 112.10337467749660333993233507, 266.08331708857917874411214143, 390.90976985716883973509538919, 329.01088679331462572008604184, 158.71128445003517981604090892, 404.59245903407168043486308306, 366.35434029559155533206649125, 194.19573855113847571374208201, 379.94898461640434561559231952, 216.17590064203761812677839771, 223.85736931998752652361872606, 215.30222433132459514126821887, 193.26277766072317376710998360, 364.27701325326768255763454363, 157.72987052894848147843731567, 326.94128157610799689791747369, 388.09280021282381767377955839, 111.18402577739000491874321597, 266.54277820804679777211276814, 346.73233678344666941484319977, 252.57711524406573744272463955, 57.01639165715037194104297669, 187.85218553479762704228051007, 282.67168803322431358537869528, 224.67611712193377115909243003, -3.14326924385573391873549554, -0.90010570311670168308637585, 96.17069437543644028210110264, 199.33582061541360985756909940, 182.27846100397448481089668348, -4.57611186272201919678082049, -257.99057047583272606061655097, -1.82557398572480522602745623, 101.69189406672646214246924501, 127.54445573081555664884945145, -4.49732115287934419711746159, -232.30379071310053973320464138, -392.35336998126877006143331528, -2.65962064013652721428115910, 64.23306507635152229340746999, -4.17823235146674676343536703, -190.62136057342786443769000471, -350.97981405658060793939512223, -367.33195935840376478154212236, -3.92525467655692494872710085, -4.21737374407493348371644970, -136.02000312436450712993973866, -288.23032697439572302755550481, -329.76997539886201593617442995, -217.04860831373792962040170096, -4.22506971989652058141473390, -72.29251053978725849447073415, -205.62211432499287866448867135, -270.03888658781465892388951033, -195.10425396920604157458001282, -3.75470502920668458557429403, -107.71214521643315720211830921, -191.64672875892048864443495404, -159.55913514871488700919144321, -2.82958247867568246647351771, -100.33839131188909732372849248, -113.28188975877471023068210343, -2.02984471265764909730933141, -59.20137760431371987124293810, -1.13006153433920530915202107, -216.94986764694439784761925694, -194.17815277072924118328955956, -366.46414265432213142048567533, -58.68817436396830089506693184, -158.58371851794368012633640319, -328.49641010394003615147084929, -389.37906114078765540398308076, -112.59894550018677250591281336, -99.47273320793448192489449866, -268.50544005644428580126259476, -349.78532743252520731402910315, -254.30643984757679731956159230, -190.57684701519576719874748960, -106.72241271551203567469201516, -286.77766766087427185993874446, -228.97129601689493938465602696, 0.65888959003758174048925866, -203.43166351223138121895317454, -71.95880175563581815367797390, -188.94812353702479867934016511, -0.54028097490086324228286685, 255.71015351091924117099551950, -135.09478864364308492440613918, -4.18373247132461756336851977, -1.97138030481194448384485440, 228.04006865312763352449110243, 391.09379142082968883187277243, -3.47586424690169337736733723, 65.01418211509854927498963661, 184.65606105019628557784017175, 348.36366144437067760009085760, 366.20261451786950601672288030, 128.93974257358527779615542386, 101.83529783301392512839811388, 283.51551025120562599113327451, 327.60501736958264018539921381, 216.03661549146309539537469391, 200.00439085138680184172699228, 96.43013782801264710542454850, 267.22322299296490655251545832, 193.49255409229277802296564914, 188.08418865096339800402347464, 57.04436608707574407617357792, 157.96464661316298361271037720, 111.31922987220724508006242104]

- list SFCorrectValue = [-0.27937980331459933, 158.30535989173870348, -0.27937980331459933, 111.↩ 95922396290268352, 58.10903859010657158, -0.27937980331459933, 193.39790162602255919, -0.↩ 27937980331459933, 268.25033740186444220, 189.82215394456304125, 0.25883372957144957, -

0.27937980331459933, 98.74870633402809972, -0.27937980331459933, 215.53653410699894266, 327.39158447388479090, 284.96744613445838468, 202.31101004349304162, 0.82513685120877511, 105.22484885820357192, -0.27937980331459933, 223.08256339289710013, 364.76793579171919646, 347.61723438903078431, 186.13386039326010746, 132.63331666423582078, 0.93045442511324694, 69.59377646537200235, 377.49981102452846926, 387.86131916037487599, 228.06934028761415334, -0.39282069172599421, 0.28340911989108974, 1.32922047073893168, 0.77036064878831922, 402.↩ 01980094098689733, 253.64580857732585173, -0.71137220262248491, -187.02281867595976905, -132.12396908339931656, 1.00935338967148880, -68.31938968731270734, 262.84597566425361492, -1.01378116441184574, -229.49291759567395843, -285.00916079306904294, -202.15395986486728930, 0.20369749822588545, -105.12807714412967641, -1.65128343645850606, -255.65451390973146317, -348.80806087323372822, -268.84160316836425864, -190.14226619109740568, -0.35372141836276683, -98.67697413471481127, -265.72733211685635979, -390.53074044891235417, -328.54385597348704096, -159.10063256399396892, -112.66830824900223718, -0.48136222438574450, -58.49296577544600950, -406.26733909792579880, -367.00125201198864033, -194.68519835904922388, -0.33966732733873783, -0.33966732733873783, -0.38739421188725309, -0.33966732733873783, -380.72973951162691719, -216.95201141250419141, -0.33966732733873783, -0.33966732733873783, -224.64681969190800714, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, -57.64885629013803481, -0.↩ 27937980331459933, -111.76778906363414023, -97.23151838524054824, -158.57787734712869110, -0.27937980331459933, -189.13083157351150021, -103.47360646666479056, -0.27937980331459933, -194.37812613590944011, -268.14533020187656120, -201.00144759920507909, -67.33160591182809185, -216.97908089605186888, -0.27937980331459933, -328.52149241710583283, -284.82532200213120177, -130.61038743255170402, 1.17408313059079461, -225.01886593214194932, -0.27937980331459933, -366.28282561516010674, -347.97810273027931771, -184.84955361689296183, 1.35575884552714854, 68.83865863464234280, -379.59834765834438031, -389.38131238244437782, -226.65542721376317559, 2.45142690491787363, 132.61870525252811603, 104.51756439557146905, -402.65102785189810675, -252.04129957093991266, 2.66640808738427992, 188.46031263508777442, 202.68330394428298291, 98.04919160612566031, -262.11224472208607494, 2.80043348997752872, 231.30365585971119913, 286.80511684477886547, 190.23232474250315249, 57.84805143771897917, 2.85708460601647829, 258.69203442119561487, 350.58867061999404768, 268.52259877181154479, 112.10337467749660334, -0.33966732733873783, 266.08331708857917874, 390.90976985716883974, 329.01088679331462572, 158.71128445003517982, -0.33966732733873783, 404.59245903407168043, 366.35434029559155533, 194.19573855113847571, -0.33966732733873783, 379.94898461640434562, 216.17590064203761813, -0.33966732733873783, 223.85736931998752652, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, 215.30222433132459514, -0.27937980331459933, 193.26277766072317377, 364.27701325326768256, -0.27937980331459933, 157.72987052894848148, 326.94128157610799690, 388.09280021282381767, -0.27937980331459933, 111.18402577739000492, 266.54277820804679777, 346.73233678344666941, 252.57711524406573744, -0.27937980331459933, 57.01639165715037194, 187.85218553479762704, 282.67168803322431359, 224.67611712193377116, -3.14326924385573392, -0.27937980331459933, -0.90010570311670168, 96.17069437543644028, 199.33582061541360986, 182.27846100397448481, -4.57611186272201920, -257.99057047583272606, -1.82557398572480523, 101.69189406672646214, 127.54445573081555665, -4.49732115287934420, -232.30379071310053973, -392.35336998126877006, -2.65962064013652721, 64.23306507635152229, -4.17823235146674676, -190.62136057342786444, -350.97981405658060794, -367.33195935840376478, -3.92525467655692495, -4.21737374407493348, -136.02000312436450713, -288.23032697439572303, -329.76997539886201594, -217.04860831373792962, -4.22506971989652058, -72.29251053978725849, -205.62211432499287866, -270.03888658781465892, -195.10425396920604157, -0.33966732733873783, -3.75470502920668459, -107.71214521643315720, -191.64672875892048864, -159.55913514871488701, -0.33966732733873783, -2.82958247867568247, -100.33839131188909732, -113.28188975877471023, -0.33966732733873783, -2.02984471265764910, -59.20137760431371987, -0.33966732733873783, -1.13006153433920531, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, -216.94986764694439785, -0.27937980331459933, -194.17815277072924118, -366.46414265432213142, -0.27937980331459933, -58.68817436396830090, -0.27937980331459933, -158.58371851794368013, -328.49641010394003615, -389.37906114078765540, -0.27937980331459933, -112.59894550018677251, -99.47273320793448192, -268.50544005644428580, -349.78532743252520731, -254.30643984757679732, -190.57684701519576720, -106.72241271551203567, -286.77766766087427186, -228.97129601689493938, 0.65888959003758174, -203.43166351223138122, -71.95880175563581815, -188.94812353702479868, -0.54028097490086324, 255.71015351091924117, -135.09478864364308492, -4.18373247132461756, -1.97138030481194448, 228.04006865312763352, 391.09379142082968883, -3.47586424690169338, 65.01418211509854927,

184.65606105019628558, 348.36366144437067760, 366.20261451786950602, 128.93974257358527780, 101.83529783301392513, 283.51551025120562599, 327.60501736958264019, 216.03661549146309540, 200.00439085138680184, 96.43013782801264711, 267.22322299296490655, 193.49255409229277802, -0.33966732733873783, 188.08418865096339800, 57.04436608707574408, 157.96464661316298361, -0.33966732733873783, 111.31922987220724508, -0.33966732733873783, -0.33966732733873783, -0.↩
33966732733873783]

- list bFieldCorrectValue = [[-0.00000466916987661623140515657, -0.00000469774735703660250055961, -0.00000395939995441359537849948, -0.00000482221010938081362973313, -0.00000396550195293026650080566, -0.00000387749965995507482504874, -0.00000382771914822608855781148, -0.00000292862967710368168300470, -0.00000309944588351366663844040, -0.00000326900335531990481345490, -0.00000454936400457388206747429, -0.00000558752101129238260083512, -0.00000408059138231045771583226, -0.00000355106400521909434142326, -0.00000435087021876521834478800, -0.00000601269263477531150000705, -0.00000422084203782920821688886, -0.00000318190138831836488695379, -0.00000410009973941543825440466, -0.00000596980094276740785646726, -0.00000437558746190408254440234, -0.00000290454751947236613300574, -0.00000382714510184938365401023, -0.00000548811576122437301257333, -0.00000452500225311201280219012, -0.00000289673824026572774027722, -0.00000355797907710025893013046, -0.00000472917096365148288520958, -0.00000464209266810378363380275, -0.00000325927840943272435566430, -0.00000331142973766262746783385, -0.00000391818849209657636467391, -0.00000209963437544594680927643, -0.00000334845606049689288255403, -0.00000442231665001654005299307, -0.00000340778215041781557426582, -0.00000496401499704663627101850, -0.00000371020589188032213470627, -0.00000647174234685046458954343, -0.00000310184058961244829062285, -0.00000741610960273363378910619, -0.00000230211586279070535689746, -0.00000754218056388060709003445, -0.00000166104789262922323326264, -0.00000688055635928999741163627, -0.00000152421125169085425935086, -0.00000570765164247120032554915, -0.00000834218389464975617795476, -0.00000857509678229675104794115, -0.00000171698141817773268018820, -0.00000083362705561108169433266, -0.00000779149266707861698196078, -0.00000061122225394470813803995, -0.00000634459329939409620016319, -0.00000132985936788230726855145, -0.00000474768270446881180025947, -0.00000294559945549423868964008, -0.00000349515548030209205720683, -0.00000506152148848857683034784, -0.00000363192483452629967890547, -0.00000705892632586430391890102, -0.00000280316637579680072320460, -0.00000484372545126640032442291, -0.00000351759447383940304743635, -0.00000278445555434836104581934, -0.00000506790710227895837522742, -0.00000360543134416433626724991, -0.00000723002305488891725015511, -0.00000270154459553407161272231, -0.00000862725220637769032246307, -0.00000151603425631581637699668, -0.00000889681845434635380212227, -0.00000054574943740535989470102, -0.00000807459544423345471292688, -0.00000028768415648587549632195, -0.00000653979801767764028344381, -0.00000104852021517869108351137, -0.00000172984723925869242351476, -0.00000083983432612446153178254, -0.00000843900811589068215474302, -0.00000767582118307908633147649, -0.00000059726856794431001312451, -0.00000625821688852544495012837, -0.00000128468751960543469050485, -0.00000469346301964793005857693, -0.00000286515300439714568582678, -0.00000347076187887664550473497, -0.00000494941956996801184741341, -0.00000363493590034705472318136, -0.00000692469606038360307309750, -0.00000281267864454127884416695, -0.00000819936244305945920781559, -0.00000317196252454694048298657, -0.00000471374717555926789889532, -0.00000371565160363171774840236, -0.00000336022805812820178159180, -0.00000617358952479633905262779, -0.00000311900978890395319081648, -0.00000710683808874491928337606, -0.00000232528792457305233048830, -0.00000725756731943215569298955, -0.00000167147731172188549259223, -0.00000664643984443812749413928, -0.00000149442711094559785073212, -0.00000553693025919358604780215, -0.00000200332775657395782884754, -0.00000431645828795938008899278, -0.00000552943336618137946936440, -0.00000513784876522024205164861, -0.00000291349221850059231320802, -0.00000284655847889421133734954, -0.00000448044829528352128109872, -0.00000310574162993285972093525, -0.00000376609468348992355569939, -0.00000367413552144735505288614, -0.00000320068755163319901570185, -0.00000441266274488435117693110, -0.00000383452939051735118357973, -0.00000510130172372361619949864, -0.00000357225423395365277440933, -0.00000551868766810142620429357, -0.00000320964682172228580624477, -0.00000301378877941614634153886, -0.00000410778685826763726801842, -0.00000388664939509098225812892, -0.00000397227444580496633736669, -0.00000410069159332783634284211, -0.00000368643251657461040024966, -0.00000424376923364454084695299, -0.00000352000049615883929109844, -0.00000437117894384989181643785, -0.00000337662784325707268512644, -0.00000444378070030683556407292, -0.00000324438745338572602120797, -0.00000442575475476003422144623, -0.00000312146444177834044445626, -0.00000430506140600200794699842, -0.00000162539700129068705292232, -0.00000201042117077484983964498, -0.00000602517698315944672481353, -0.00000574081659120288884194097, -0.00000247517126240858007990759, -0.00000494816587723671617155257, -0.00000282491005183331607728903, -0.00000383106878716190527245583, -0.00000410633847534577906126596, -0.00000270729373226552707089668, -0.00000462023025166119135509089, -0.00000188570036238920448643473, -0.00000526327669295933915524733, -0.00000153163976212105700181127, -0.00000581066380738313498772737, -0.00000421442728380309715322319, -0.00000504756703827697536253027, -0.00000170989697017502109315835,

-0.000000368205651246759689274 25, -0.000006104964874039769308507 71, 0.000000147970715399805712407 52,
-0.000006998154097385826055103 14, -0.000000139543256453925005840 17, -0.000007325311268714011280823 22,
-0.000000957449086117795926051 14, -0.000006814847503880925708760 32, -0.000001919098645207326594053 39,
-0.000005462538042366831693268 4, -0.000002659382888561019648448 76, -0.000003582988164699999556671 26,
-0.000008127960838667446259319 00, -0.000007457537766348704991409 37, -0.000000234238167987601168211 21,
-0.000001530497702892756555463 50, -0.000005735090100133847753390 71, -0.000002538748528520541344422 14,
-0.000003362471878848714567928 75, -0.000004277978857843769559853 38, -0.000001009160251116196961387 07,
-0.000005313268141089583862588 92, 0.000000665965933245403488290 42, -0.000006633416101256644790214 35,
0.000001286449913954700983257 92, -0.000007741212825529410768436 90, 0.000000871395950407652559332 63,
-0.000000686976806836211766085 19, 0.000001087106470070684856888 5107, -0.000005372574302659216289191 49,
-0.000006762387901441110924850 07, 0.000001735673372247141482926 88, -0.000007920443808640041069872 24,
0.000001272895962333544302592 24, -0.000008304913740411444737061 18, 0.000000062266992184529590216 48,
-0.000007564410539416802209118 47, -0.000001360573441233412476806 04, -0.000005713940709338972224811 89,
-0.000002477620552432487205092 41, -0.000003185525822041595355317 13, -0.000004285070874239019008832 17,
-0.000000099761910657629347219 19, -0.000001427228406237246719401 23, -0.000007121890788624390749520 65,
-0.000005403874044001729978867 37, -0.000002481624482490750735133 46, -0.000003073674819372230883302 93,
-0.000004232310602584853629661 02, -0.000000777199925254534555820 50, -0.000005212217344728846509167 07,
0.000000853268705209387845752 14, -0.000006467267976458186151970 85, 0.000001450366431284919655866 67,
-0.000007504937291090763591802 55, 0.000001022957800677187251607 08, -0.000007828939574459940902730 58,
-0.000004852333446753194925538 20, -0.000005787573750391326826518 75, 0.000000020287080273935243494 07,
0.000000494123708124895733633 46, -0.000006552513892308689661726 51, 0.000000176526386444774723423 77,
-0.000006767676752540858455193 30, -0.000000683554916541925163217 55, -0.000006192424092432093462169 51,
-0.000001712915954191518598656 24, -0.000004844962781809541334590 57, -0.000002546711824114191281778 44,
-0.000003032326826771812897331 99, -0.000004125421682941469509386 60, -0.000001249268188050931304157 69,
-0.000004900055231256016360420 24, -0.000004100792051189442145528 19, -0.000002167802257612540191332 80,
-0.000002659935093759879472431 36, -0.000003044511846408271801877 12, -0.000003978446178241522450977 18,
-0.000002008865214462643643099 23, -0.000004343371960747792585877 13, -0.000001264973063473134013685 40,
-0.000004820345137294431170111 19, -0.000000970477087679776405311 54, -0.000005199108208945110929364 61,
-0.000001123954920169712695175 01, -0.000005269881142691022743776 95, -0.000001590297779107672852580 53,
-0.000002790076419657463466597 34, -0.000002707920534783716912688 40, -0.000003705861395300735868418 76,
-0.000003639831032048499611622 66, -0.000002670940549635263427431 54, -0.000003547311581062633304968 58,
-0.000002674698768215354493561 50, -0.000003419714389657307361184 00, -0.000002717673792120109748461 43,
-0.000003259265982187504637309 5, -0.000002801549574795857157024 32, -0.000003083134575019942040502 83,
-0.000003811687890361988188879 34, -0.000002918682155814867151366 44, -0.000003758574669448968346373 82,
-0.000003275371435838905081349 87, -0.000002948112853648078168658 60, -0.000002431318723147511385679 30,
-0.000003126679961502833175648 45, -0.000003648708365475050383420 432, -0.000003827816526512429553283 45,
-0.000003183129155410189311398 16, -0.000004313944058749856086479 20, -0.000002607368642943563552788 04,
-0.000004447290998259228003187 95, -0.000002102292897194702951088 65, -0.000004223755108950629635965 02,
-0.000001848759477094594787840 43, -0.000003767219061504040205539 19, -0.000001960451867220521750473 98,
-0.000005923420840851475748943 45, -0.000005544760707924370150411 36, -0.000000814003347248115565701 91,
-0.000004253244226850051261108 3, -0.000004699591704416164490286 81, -0.000007373456996621571907577 1,
-0.000003754035505591927710645 04, -0.000001737684900976831395982 64, -0.000003076109720272052050648 72 847,
-0.000003165602208531349471594 28, -0.000003512825164681814800536 19, -0.000004596921397320425354713 83,
-0.000002701527771238801390441 38, -0.000005604500464185683460084 31, -0.000001687199993214851171240 79,
-0.000003423634737932706773077 05, -0.000003165555161410657517936 450, -0.000003199467932657861860961 40,
-0.000005114769646552445833177 52, -0.000002384952630150790012382 97, -0.000006470135878608972665752 74,
-0.000001082573787397974524103 57, -0.000006914685681239927260075 24, 0.000000031670643635583581195 60,
-0.000006434154519761684247382 98, 0.000000507715209298222229401 54, -0.000005329878432634424524397 409,
0.000000062410153394725731244 98, -0.000004080209874444685043360 86, -0.000001287619385107277441640 81,
0.000000305300963399226811309 37, 0.000000804931633721443997277 14, -0.000006759106531276178780106 44,
-0.000005563573402434483275143 34, 0.000000310536619466591689908 48, -0.000004204891421702167735529 29,
-0.000001158304945279706829906 53, -0.000003203034977046413815985 30, -0.000003231981014917997459140 46,
-0.000003394080338639974761408 63, -0.000005310010027204825310540 89, -0.000002280563636248469533586 57,
-0.000006784693242381760576659 77, -0.000000884776552097538224539 25, -0.000007274133226560342348231 74,
-0.000003268978054311831541658 29, -0.000005161678927616114361952 47, -0.000003428506799238336935137 98,
-0.000002404325456694791921576 65, -0.000006506812171691695135985 97, -0.000001125054147143240300041 59,
-0.000006951900538091945776892 31, -0.000000038109380746886272567 05, -0.000006473838782693360876770 51,
0.000000414566471161729396374 99, -0.000005367828254428663770656 07, -0.000000040386984611623721950 87,

-0.000004110903410176831024986333, -0.000001380732056537456377710 15, -0.000003183728073491490218310 11,
-0.000005627275287447657896560 80, -0.000004776581892182495043305 39, -0.000000612278458906465059219 61,
-0.000000948270888847624863399 24, -0.000003815478480871763777087 63, -0.000001934438735728849900871 64,
-0.000003112106362082083816687080 44, -0.000003318697971439304780631 21, -0.000003521997951986014315948 69,
-0.000004704946597670969516812 73, -0.000002738495074569903078875 49, -0.000005688898621621768605146 74,
-0.000001769528991031094350896 57, -0.000006004517856198477860068 13, -0.000000951497156201585765080 38,
-0.000003661100896278850712182 60, -0.000003234185591876674678596 00, -0.000004026008636874376506436 38,
-0.000004469495847771409014577 08, -0.000002724001468378236545539 67, -0.000004586721590835389027304 62,
-0.000002302351486545032412730 11, -0.000004355930598681501865628 54, -0.000002128833052767958072446 46,
-0.000003885539920361719780144 90, -0.000002287528065415136106905 07, -0.000003367620965454554092580 5881,
-0.000002750529684441294707799 61, -0.000003001205317270282573659 20, -0.000003390656970543113683374 27,
-0.000003914326068653997588988 81, -0.000003867189823242919772985 74, -0.000002837600839175084593070 69,
-0.000002840604460088215984267 35, -0.000003720199167780288467612 42, -0.000002870573199346581707468 64,
-0.000003490914196441135331953 16, -0.000003825832351644726756509 46, -0.000003245016470337282255470 34,
-0.000003818641803553960688288 62, -0.000003048967944455735619121 21, -0.000003847908125245524118163 58,
-0.000002926265125261060909717 68, -0.000003892308380863364904416 87, -0.000002862738197587269642238 56,
-0.000002492694581753588705861 06, -0.000001655482237209619174163 01, -0.000004406155336826438219778 94,
-0.000004975340784626303956589 71, -0.000001287683300810760832579 88, -0.000005486593961727192739652 89,
-0.000001385756011294490192730 30, -0.000005706228818302739558515 45, -0.000001801015181645569025002 26,
-0.000005457598964741178506026 97, -0.000002320718332930050728204 12, -0.000004707444945448636833360 69,
-0.000002743208692611579841867 32, -0.000003616020659077754512405 33, -0.000003992667533797232255841 22,
-0.000000943664169711307778902 31, -0.000001893959634366278747664 54, -0.000006824526704582180925199 01,
-0.000005569244589370603003308 27, -0.000002641951482763364134017 43, -0.000003753790329346908617768 12,
-0.000004138060931600643000847 70, -0.000001891613888080865604464 45, -0.000004910989905867787144321 15,
-0.000000511858972162371407152 39, -0.000005940090633514393263399 37, 0.000000066869698384312723880 37,
-0.000006849347666819941143358 76, -0.000000162249097066521752707 46, -0.000007239504929639869315930 42,
0.000000191391818799561805340 25, 0.000000914529967001440018990 81, -0.000006601026404545618444402 21,
-0.000007780046901856716898802 92, 0.000000606924975258649377351 26, -0.000008288517969195862596641 12,
-0.000000407243207521755645005 55, -0.000007768517864930579020074 29, -0.000001632687759934762493114 08,
-0.000006184252258447503232798 11, -0.000002585877388757879148791 99, -0.000003894962601007848762828 95,
-0.000004241965490617003539544 71, -0.000001547524222414936022091 27, -0.000005260361892204286456337 70,
-0.000001584287326956026389354 14, -0.000002587251259447109409157 64, -0.000006477356464652411522243 26,
-0.000004041447503927193620506 91, -0.000004290813773417610484193 36, -0.000001538564664527889685189 31,
-0.000005405471948075501606651 49, 0.000000322772421190406936908 86, -0.000006863814565985649504911 44,
0.000001105656232092042497957 9, -0.000008145016782084720281136 88, 0.000000788619495527595908613 274,
-0.000008703714485079735570737 49, -0.000000285107883780583305326 62, -0.000008159446728336942360648 89,
-0.000006695377111494688383039 16, -0.000007898137635483467843271 37, 0.000000591116908228940169047 717,
0.000000339588975320378274061 43, -0.000008434503701231834684685 22, -0.000000607310206707123813079 03,
-0.000007954749258240881776066 06, -0.000001763608810331004089839 62, -0.000006426076931872221634440 07,
-0.000002649509538613160246274 84, -0.000004197271373554537065978 83, -0.000004279506244418695642766 32,
-0.000001893290769712060806547 35, -0.000005328855800032962807867 30, -0.000000162181045123928933213 14,
-0.000006056476665462023338158 07, -0.000004357756855900262932582 67, -0.000004212102411765054810926 29,
-0.000002766586451897782492737 69, -0.000002576634559143457449306 28, -0.000005046894628565875934539 47,
-0.000001207757487523629997183 60, -0.000006128723680501494967451 90, -0.000000567284229447634383124 02,
-0.000007087422650970049133544 41, -0.000000685820997888661863438 56, -0.000007535507417486027727064 61,
-0.000001335510027280870001933 95, -0.000007201997199238423191618 42, -0.000002150389688223353754537 20,
-0.000002208641245363017065433 78, -0.000002144558714932919259084 58, -0.000005846995424213146957286 62,
-0.000006157734472189378818534 83, -0.000002368629268134221628152 50, -0.000006032276338722278724309 00,
-0.000002692053956048653862631 52, -0.000005441910208662434097310 56, -0.000002923694335091058528984 78,
-0.000004515210563754727400337 56, -0.000004101112533974774877287 93, -0.000003501150522541799881774 25,
-0.000004607095381496415074809 67, -0.000002671269673319149039756 11, -0.000005257454871785676784440 82],[0.↩
000002366897358546120360344 93, 0.000054013542118938742209524 61, 0.000053612097522474058588614 48,
0.000002126343868342701241505 20, 0.000213763439098918608732499 69, 0.000216019332457135852613754 946,
0.000213713517162110450261717 11, -0.000213003594260546147030541 13, -0.000210615828168400626282594 22,
-0.000210617417987443315613091 40, -0.000049348730125814980118290 07, -0.000049411245329527477151108 92,
0.000206397270217370624187727 68, 0.000206239154234050270104311 45, -0.000096773531865318148173504 83,
-0.000096692041844518845754297 14, 0.000192744965932244789914851 16, 0.000192447502414842309698822 52,
-0.000136642271729642527647961 25, -0.000136485020894412602439349 50, 0.000171294275122681557586498 68,

0.0001708668769204698192898384O, -0.0001674012377176852092632714O, -0.0001672418319567748950724150,
0.0001408417614258951290531068 9, 0.0001403372085214228204633202 2, -0.0001891472586420130770016451 2,
-0.0001890378516932557156557392 9, 0.0001012468443445306256122870 3, 0.0001007491016906802435522047 5,
-0.0002030535688162263209446262 2, -0.0002030100474241894629653937 3, 0.0000996428817067849679958019 3,
0.0000528318299089354878215682 4, -0.0002027627288044380176462722 1, -0.0002105632296589985883155521 4,
0.0000018356746326187827812068 12, 0.0002136030643854813565116201 9, -0.0000491922004299690619278165 3,
0.0002058621149947041770091743 5, -0.0000960896626116855258553997 6, 0.0001917256935831193078150630 5,
-0.0001357160796902891165480187 2, 0.0001698353679896720622860550 9, -0.0001665290045950697445705651 6,
0.0001391499033986071749615515 1, -0.0001885350328111459544070532 8, -0.0000950914845506494566884994 6,
-0.0001344776702388088772920082 8, 0.0001906822404175828838783501 9, 0.0001683457928036513433483339 9,
-0.0001653886881119072263870983 8, 0.0001374435840651389234903789 7, -0.0001877267150990975970448193 5,
0.0000980709809474170058317399 3, -0.0002023561789048729838823986 7, 0.0000517568102416531461602844 3,
-0.0002104648458379298038914417 8, 0.0000014989218848652966017563 5, 0.0002134467315430453810201627 5,
-0.0000487643853443974595969143 1, 0.0002053190727422915778498796 5, -0.0002018586041968212002519628 1,
-0.0002103403719384233064478018 1, 0.0000505310916096229039004214 5, 0.0000011415490115152406263280 7,
0.0002132659547464422304434950 19, -0.0000482214694502303487032117 4, 0.0002046879543817260466848084 7,
-0.0000938757035663368821342550 4, 0.0001894691987245724132653962 3, -0.0001329790500530517660063983 8,
0.0001666165999243700204107687 7, -0.0001640093044836001521819685 0, 0.0001354679487044976469913709 6,
-0.0001867459688588596377397527 1, 0.0000962590693682132146407823 3, 0.0001882645188932185391188223 2,
0.0001649043048671920978949512 2, -0.0001314588428333635539580620 5, -0.0001626104298294419941636745 9,
0.0001335192805923695365366209 6, -0.0001857498336583359940538812 1, 0.0000944816580162036103330419 6,
-0.0002013511752732333767550165 8, 0.0000493412437479562069144768 7, -0.0002102115663846385951521456 0,
0.0000008157908071576898908664 8, 0.0002130857834614150033911150 8, -0.0000476524059069439992865693 8,
0.0002040599105096380149002716 1, -0.0000926369455495397469855720 1, 0.0000483746683585351165885792 0,
0.0000005815570103914553045529 0, 0.0002129311602092056547163440 0, -0.0002101004366004537995352846 6,
-0.0000471346942825844872067102 3, 0.0002035254907090042197914669 9, -0.0000915597676612431212589404 7,
0.0001872442275766810632076087 0, -0.0001301499138369818656460474 8, 0.0001634613423998361307100390 9,
-0.0001614097597201151587827106 0, 0.0001318881314896674971162693 3, -0.0001848954518591366122728936 2,
0.0000930087927689131663917462 8, -0.0002009155763653248754057462 1, -0.0001292527053943813900287879 1,
-0.0001605935033416280876585047 2, 0.0001624954286037077772625325 5, 0.0001308080916011625531494150 2,
-0.0001843171298749747652901548 3, 0.0000920503488216943224374186 3, -0.0002006216565801628812660412 1,
0.0000477686240935890033898828 0, -0.0002100258190941126275070455 2, 0.0000004709237996336846716660 7,
0.0002128232222028765465576816 1, -0.0000467464512956048861718784 4, 0.0002031610171369804894014182 7,
-0.0000908071106976631532876898 0, 0.0001865553871302523937601020 0, -0.0002100004573901620766578773 9,
0.0000004684967229754687254347 1, -0.0000465708949745120446769841 6, 0.0002127761835919622136498868 8,
0.0002030172846362415762459102 7, -0.0000905086596896974624306200 8, 0.0001862945598901749827500801 1,
-0.0001289125994940812911827843 2, 0.0001621393972344384315498189 2, -0.0001602915307542482640981024 2,
0.0001304210879058626600546927 4, -0.0001841069347192423981407483 1, 0.0000917214482719311697666436 7,
-0.0002005169107198801002178018 0, 0.0000475804229043332518205890 2, -0.0001605577198161971003587816 8,
-0.0001843008474212557605900753 0, 0.0001307662009971822375791611 3, 0.0000920415908513251231938545 6,
-0.0002006192339004583531734266 0, 0.0000477990117158656036390820 8, -0.0002100290145121768109297277 8,
0.0000005325713039769806909587 0, 0.0002127952643891775413517719 6, -0.0000466708502500167877612320 7,
0.0002031122197053476677395395 0, -0.0000907351781385777866060019 0, 0.0001864947348999488588219136 7,
-0.0001291963553270634957811924 3, 0.0001624358422296066396806058 0, 0.0002128759703596994502100558 9,
0.0002034284213069365943125227 4, -0.0000470515861692508881590664 4, -0.0000914702013127293887210372 8,
0.0001871215251246879278726464 9, -0.0001300734923174950118107684 5, 0.0001633349369255839424013021 5,
-0.0001613586684156551097433368 6, 0.0001317847128220592158726054 0, -0.0001848724181888910286352539 8,
0.0000929526435571405756962473 6, -0.0002009139952759911719286367 6, 0.0000483768587989016139836069 6,
-0.0002101073204549225119635441 6, 0.0000006345594358553811461265 4, 0.0001333291043674130513067843 7,
0.0000943295938650784350932437 8, -0.0001857355376092607618810498 3, -0.0002013559657971078552338856 1,
0.0000492422354249527627070713 7, -0.0002102229766093298988019477 7, 0.0000007716105742740120645395 9,
0.0002130048287550869454479163 2, -0.0000476500837774190158058033 9, 0.0002039158287297398769628842 3,
-0.0000926034419799783400017653 7, 0.0001880782234053784225582400 8, -0.0001314145663441098732093087 5,
0.0001647016302953023406466237 5, -0.0001625754412958852856800617 8, -0.0000483582405427815552518590 3,
-0.0000939527127330402005546752 8, 0.0002044991607630001020402821 2, 0.0001892191542983394510982903 0,
-0.0001330108539236138690507738 1, 0.0001663314941852682860446571 7, -0.0001640203078420876417562596 1,
0.0001351746309401440743983707 8, -0.0001867566797129130147341047 3, 0.0000959812012743714522465879 3,
-0.0002018759617929509605702143 0, 0.0000502870667485641969053980 5, -0.0002103572353074076653478563 5,

0.000000945831546361488028777700, 0.000213161468552725143861190009, -0.000187775075555341548413923470,
-0.000202391314814704039626090909095, 0.000097661762708549853606934680, 0.000051361120933515946256325631250,
-0.000210487880383046309764119570, 0.000001142268415573614673906690, 0.000213321737595180828221674840,
-0.000049055358653510537703756460, 0.000205089184217387615395505460, -0.000095300402920074772108137430,
0.000190370555610442862169359720, -0.000134609317898834126734849970, 0.000167977443041761651367285710,
-0.000165465818928934839610392360, 0.000137043352496085294811925540, 0.000205596273242615845125089160,
0.000191356880560719828671817360, -0.000096429345734682237866602920, -0.000135953323875821389206050570,
0.000169387099236767652722736942600, -0.000166680613984285935780249590, 0.000138646318272511482076400640,
-0.000188627633186269226452796980, 0.000099108713805188419386157930, -0.000202818616221805362448415980,
0.000052295007860146025980164350, -0.000210592676573964068462152530, 0.000001330359001511941881057450,
0.000213461389402232079538432320, -0.000049628667999307527359392770, 0.000100095668511400949962815430,
0.000052942045806407151960042150, -0.000203086748763526176076971652, -0.000210652858558419564397393150,
0.000001482797231078990299573800, 0.000213559778799039438152915230, -0.000049981868773111537396746820,
0.000205944208740127212462290310, -0.000097152928638991490413630290, 0.000192028477701103756367442820,
-0.000136822024657846930129068140, 0.000170344496869658069757405920, -0.000167465770306396844867985460,
0.000139735285622910159744267110, -0.000189174562319515023045557460, -0.000097340819937043174584455350,
-0.000137064165602774170379246630, 0.000192285509876484640582119940, 0.000170706340480553461639093290,
-0.000167686274376499988483096850, 0.000140147361084074802270102440, -0.000189321596988784080345396134,
0.000100477298597832136271140730, -0.000203148151925787876304300983, 0.000053212739632084987272554050,
-0.000210656118694941364493863900, 0.000001590473867554816558511310, 0.000213603000323906870626736180,
-0.000050042845332859443050008840, 0.000206081987010186276846902610, -0.000202986512863565650001382884,
-0.000210598625469318365653614840, 0.000053087264801852376491786420, 0.000001661969305362335990822400,
0.000213585979126585659079728960, -0.000049776611175894935504840200, 0.000205991687493712791371853450,
-0.000096940750199499544173011020, 0.000192093693335806091088829860, -0.000136623010183028434664717630,
0.000170424908901014638719362270, -0.000167293129211993181959181040, 0.000139830959006838372645636910,
-0.000189035332686850177357634760, 0.000100211685893732794583668720, -0.000135548222352932933954524406570,
-0.000166332456678990107902180170, 0.000169554305763017427229749630, 0.000138850911705568015510475770,
-0.000188349098960462744446753280, 0.000099361363228582289018955830, -0.000202619621994215077427505208200,
0.000052610641167274193718411680, -0.000210485749353153150360498390660, 0.000001713373908990238250800590,
0.000213513193711971978451448663120, -0.000049198502421013869513686380, 0.000205691116958947478708881992,
-0.000095992045901572322893005600, 0.000191489246002345131362001870, 0.000213397915609302287962001370,
-0.000210331375162080074728981890, 0.000001759661147329894187042840, -0.000048377218822747961219938460,
0.000205230849964004894175126690, -0.000094623227002166648770715980, 0.000190572600814000511076917730,
-0.000133990044805612300814201540, 0.000168240309297866292171513840, -0.000164939163113440460640371963,
0.000137375113218885755743056690, -0.000187358237578516583551196950, 0.000098078844148715370840174140,
-0.000202096910509959120337553080, 0.000051879967597000721404951890, 0.000166696266829799416954796530,
0.000135646001884513387435171610, -0.000163315545841485757275202140, -0.000186205412421468961782153320,
0.000096580198229958882905586690, -0.000201492037785850974058662890, 0.000051027596034635105423450270,
-0.000210155903521593029207106820, 0.000001812514518534049349582510, 0.000213260092342692440239501630,
-0.000047422441815262537782339760, 0.000204686190012896135608649880, -0.000093029987971340730905714380,
0.000189492186510155824659407120, -0.000132174604818487975103208370, 0.000001881830967338217209180460,
-0.000046463061782277982192145440, 0.000213123201870917016615481219010, 0.000204145365359041663661648910,
-0.000091440598568857326799291800, 0.000188421074215556583024422380, -0.000130366382635508000226859380,
0.000165169327784489902840840450, -0.000161699104174748103761316550, 0.000133942039318304430273176560,
-0.000185058535695376506173820742, 0.000095110987885419588368056400, -0.000200891725191817714238771211,
0.000050201618022458597309469470, -0.000209983244621822945667408140, -0.000160325974878641557069290140,
-0.000184085069267366890606932310, 0.000132535335098863592219117780, 0.000093909704118956538725775700,
-0.000200382615170496310033967920, 0.000049544776772159789782375150, -0.000209837266872912914978733944860,
0.000001975788217726305476766970, 0.000213010571071211401734671308, -0.000045626277807974468425087207,
0.000203695810707373215929158470, -0.000090080220439374251734793160, 0.000187530141738505090002325780,
-0.000128827274870695870955319970, 0.000163902366166760463583526590, 0.000212941725299524724137473220,
0.000203410651098912827845419100, -0.000045021660043095017213633100, -0.000089139889721222940349200370,
0.000186961818593592407535680660, -0.000127778716838834055253421030, 0.000163096946131507041606242820,
-0.000159395770324512322602475420, 0.000131650597816238950623843330, -0.000183426777776144828363893980,
0.000093171970162796056252578220440, -0.000200038204462154927884554030, 0.000049172351298837292579019250,
-0.000209738244585956282303382260, 0.000002097473806906583059000070, 0.000131430591197171278809004942700,
0.000093019597210494924922470518100, -0.000183178584750789009288896030, -0.000199907836282025381993004980,
0.000049152063930523292354033800, -0.000209699844245720898316953650, 0.000002240577238195626920614540000,

0.00021292932978375189549680169, -0.00004472865265418797314431187, 0.00020333744884580098898942124,
-0.00008875081727962814482409820, 0.00018680831799521921822264392, -0.00012737006056659394451040235,
0.00016288256973613317985481730, -0.00015904222592714236392610894, -0.00004478461487772766256567833,
-0.00008896441909145279562567271, 0.00020349091060582237284332097, 0.00018709768610017716130092258,
-0.00012765611819718898720804612, 0.00016329711683072274100089749, -0.00015931336728019438722311896,
0.00013191433264510263648726374, -0.00018337456214367854005070635, 0.00009348155509948895258969664,
-0.00020000939188601177557910726, 0.00004949306625213800757759341, -0.00020972710623493162466514372,
0.00000238983050735444537608729, 0.00021297718916038889478553608, -0.00018398315763743915704844789,
-0.00020032675406499094263902683, 0.00009449216034864956273442438, 0.00005014870894073889949667244,
-0.00020981572132553852100440850, 0.00000252865124875403699013774, 0.00021307960767526475998924484,
-0.00004517421321322968294281405, 0.00020385063315851497875441645, -0.00008974154212259016652562110,
0.00018778976759760101469982907, -0.00012858803279102554682766790, 0.00016428179261177328789912677,
-0.00016016452372189310328527700, 0.00013303259613173861407829646, -0.00009095873254367217101545168,
-0.00013002181463364067626545251, 0.00018878250468945596720997393, 0.00016569080202047639280814317,
-0.00016146599517999967333264921, 0.00013462003300271314703770442, -0.00018491239573189665462704667,
0.00009590359714470447229615890, -0.00020081238310014058756104716, 0.00005102676789396104635714718,
-0.00020995270643397084863249202, 0.00000264452492808631948819384, 0.00021322220444996276391205758,
-0.00004583031933930512133099078, 0.00020436418097369337125608635, -0.00020139457054747816238185476,
-0.00021011837168406554765889838, 0.00005199824380140341512730393, 0.00000272280146821860843779368,
0.00021338405786834859704896583, -0.00004665295441673965732816767, 0.00020495498505802614542213591,
-0.00009243352129873707472614702, 0.00018992737085119523830073485, -0.00013174350693034869328942871,
0.00016731337563349006755317183, -0.00016302382379238214318677158, 0.00013643967874164272336204307,
-0.00018602406973527294193650194, 0.00009750611236171419617460576, 0.00019105156639495987505941721,
0.00016890490087334553408887627, -0.00013350205524091196627317624, -0.00016460890798362668775191298,
0.00013821707486982872888948981, -0.00018715448846857901167052562, 0.00009905634124965563037934491,
-0.00020198824848110914710781760, 0.00005291133229075066968152136, -0.00021028927693523695495805836,
0.00000274215455054867476503079, 0.00021354085513318514727779118, -0.00004753000996580814083592587,
0.00020553385861102628408975113, -0.00009395478661697239395125814, 0.00005361925186353092243020033,
0.00000268632338141147337766401, 0.00021366857053165268470577842, -0.00021044173118879013839946368,
-0.00004834422379271559745261139, 0.00020601242450934444377236288, -0.00009530652655494274011482858,
0.00019198380347336301888529209, -0.00013504261160062464952429095, 0.00017022320317893461409999856,
-0.00016598979521232747752490511, 0.00013968135484984135763486535, -0.00018813872729791416139703830,
0.00010031552524288051128284471, -0.00020250774239127889504892555, -0.00013614143289071532218724447,
-0.00016696641590870862350767800, 0.00017106461539639585727259896, 0.00014060608245580221136589172,
-0.00018883521983308152467420649, 0.00010108788072778741536951463, -0.00020287959883068579680816257,
0.00005400895300013944418104916, -0.00021055530532653992238471818, 0.00000255675348328319715756067,
0.00021374712003927040580679442, -0.00004898045466993483502190360, 0.00020631647121319583097591444,
-0.00009629314020292675021378548, 0.00019257997464974666631393196],[0.00024221422419563195757054308,
0.00023512113755097093967770527, 0.00023058057863197106425283411, 0.00023748759163156555468399134,
0.00003642079935919566400105013, 0.00000004876154627997606448055, 0.00003573772057353782275867377,
-0.00000002088286326493736915215, 0.00003643061773194065450709109, 0.00003569517911954019377326897,
0.00023527044183071184400672660, 0.00023063332513228700505585056, 0.00007358337887977286866352661,
0.00007220446304952278591526155, 0.00021542909687234535072962449, 0.00021114630243880184068579198,
0.00011169629608104612384159027, 0.00010960034796322819868226450, 0.00018572736855210116978937007,
0.00018200969595762213548270765 0, 0.00014977057135253255752520285, 0.00014694921814450832191559226,
0.00015003603417210176467627702, 0.00014701794287069622430355609, 0.00018543487975514105141165710,
0.00018191940662472020034771958, 0.00011189019276786302020849917, 0.00010963275076850091259601794,
0.00021517429985181928792613293, 0.00021106024557111693722155632, 0.00007368546393960541179488088,
0.00007219722349858864133415048, 0.00019882741133448443050790733, 0.00021717139964537093962601566,
0.00006793431900729370216413666, 0.00003358734532889100704361116, 0.00022362899337138533338228963,
0.00003368256367660931566675175, 0.00021713104697561533816020651, 0.00006805053168791414203251527,
0.00019874982164681107205984889, 0.00010329023768253383329385198, 0.00017129906074155973219434734,
0.00013847623746089191281057051, 0.00013835141970931563249261331, 0.00017140640187023446923929304,
0.00010316280799831033482098774, 0.00017871586924658074677953190, 0.00015400894491171836209707224,
0.00009300516107585952724906564, 0.00012467183885985752599470655, 0.00012437208778106738924287833,
0.00015429306820766521858578901, 0.00009273112439351300425195018, 0.00017893891052621962111345011,
0.00006106174678821117555223980, 0.00019540257626958067605632552, 0.00003018844253601195144168408,
0.00020116622036635067394608201, 0.00003033399547645973360981107, 0.00019527973369703438663753936,

0.00006127981404725383425199925, 0.00005184462628418332080142486, 0.00002562935004478196328249646,
0.00016610642401736482411979401, 0.00017096096314956807891982660, 0.00002582048400143181171952945,
0.00016591982488141181466961838, 0.00005215152293647389768410996, 0.00015181650224050684456524041,
0.00007913788028676893213358284, 0.00013080666634865718172577032, 0.00010606201718466357910262976,
0.00010561992236396534970265665, 0.00013123095997358502845867023, 0.00007874056746028607154264806,
0.00015215293565708012404023852, 0.00006222040683111706386304585, 0.00008335959549927594771506767,
0.00010258632797025758385221772, 0.00008281781030143020561583300, 0.00010310196836561219915252841,
0.00006173032279876991220736449, 0.00011949298696466503010453153 9, 0.00004063789668697304286251737,
0.00013040326174395598201986857, 0.00002008545666738634850921955, 0.00013417078693930139946313695,
0.00002031544022311951151228693, 0.00013017928887589910106996338, 0.00004101619549416230968083563,
0.00011908714891399693085790451, 0.00008966129511665799339120708, 0.00009220505754501783900883849,
0.00001403056240262716242129474, 0.00001376989274697555899675417, 0.00008942866614871386606554216,
0.00002830234007726331423157248, 0.00008178486589571475084449459, 0.00004290405945425181214207197,
0.00007043310183359100635439726, 0.00005743847000855366427595275, 0.00005684299892959915643598015,
0.00007098765321529926426193102, 0.00004235476262650960586854040, 0.00008221259424061854988261683,
0.00002787253843428859665471917, 0.00003558242376847433067945520, 0.00002869342762609467504695757,
0.00002930084053037463522202975, 0.00003613072621715948874081617, 0.00002135823672786075472077895,
0.00004175360325406188895941417, 0.00001403897976335879906484988, 0.00004545334004177860255831167,
0.00000692532427039974761851914, 0.00004668015966986112082279942, 0.00000720767128861170651557338,
0.00004523500843280730523889716, 0.00001449990051279543850531236, 0.00004134314658568343837669740,
0.00002193448472471466867070311, -0.00000018536275476188765807097, -0.00000063555088312147990051593,
-0.00000069105112124982154233108, 0.00000010937065147812057006363, 0.00000014112585398400017200403,
-0.00000067819484557037945251883, 0.00000012223552315333760713112, -0.00000062520755138185074585962,
0.00000003733967313262340997177, -0.00000054985479003812424010330, -0.00000011406121089387746799567,
-0.00000045335505234210048940157, -0.00000030872975031862248642604, -0.00000033173271494498099654726,
-0.00000049961619875163898049847, -0.00002976391214334230916720159, -0.00002224332488489791864001363,
-0.00003633824795277500798460282, -0.00004233538267895286426930473, -0.00001468822620388765633799976,
-0.00004640416296459308508667169, -0.00000728913626897230496049212, -0.00004789692346398960015292981,
-0.00000699111979228816111486648, -0.00004656460671813850821752767, -0.00001422027357655865031330240,
-0.00004265386185347319115447606, -0.00002168971741222897682468268, -0.00003679552505313648966224147,
-0.00002921817133760940435696120, -0.00001382053872145808158609231, -0.00002803080423196321196988023,
-0.00009060990761493040927636311, -0.00008296424221490546820501760, -0.00004265961777418032834169673,
-0.00007153758346569133279302388, -0.00005733533440305722567609942, -0.00005782780795828509516143603,
-0.00007114058714583946195497699, -0.00004317621071903213610066949, -0.00008269851606185571078408425,
-0.00002847984121712703668954751, -0.00009048023917856811966595881, -0.00001411327712725095190880210,
-0.00009327121660228933602409390, -0.00010318331096309616562445299, -0.00011984926043979200348168351,
-0.00006244976420704460807684089, -0.00004117762498992887174997096, -0.00013103893622691932877022403,
-0.00002039581194293668963161623, -0.00013502106123139932542659425, -0.00002011614821639516964232354,
-0.00013114060303592856016052792, -0.00004075872687014014603135159, -0.00012006568355820456989353773,
-0.00006197987877876467933870724, -0.00010352074553398723177032037, -0.00008323192631180004077799672,
-0.00008366647472881513195215597, -0.00016661982395304531716188901, -0.00015254720284414819103484262,
-0.00005191459148860947298053173, -0.00007890803977945857377099270, -0.00013152454305630307466838669,
-0.00010591450193005250970598730, -0.00010629131245839624965877218, -0.00013124084510660762969806725,
-0.00007932537509100781797973662, -0.00015237219495052033743469344, -0.00005229449907647272581491180,
-0.00016654075739959696661300625, -0.00002589553811933865517776535, -0.00017156478127045330598493544,
-0.00002563586837672157138761503, -0.00009315588308157499923371769, -0.00006140378828223462081940809,
-0.00017902977521352931135692477, -0.00019563886207072810237417304, -0.00003040125774207811000515722,
-0.00020151795372538009896642985, -0.00003016758948722044825696206, -0.00019570217940511938615154308,
-0.00006106996966507975167781641, -0.00017917336493616733169957633, -0.00009279481635922343226927089,
-0.00015447997982178347862840251, -0.00012451474238563082835504203, -0.00012483612445501407950051997,
-0.00015424205974566204940118253, -0.00006787379414957882630982794, -0.00010310846865362845376553474,
-0.00019892317488940359734329866, -0.00017150610400596397794176218, -0.00013832132833937183760859846,
-0.00013858878543495013356198764, -0.00017130791711928515627834380, -0.00010341001070076125513755305,
-0.00019880316650244890031885847, -0.00006815575615437242427900690, -0.00021722025818499458046755002,
-0.00003373987825875695495486546, -0.00022373374742836631742098907, -0.00003353732141415467825438373,
-0.00021727331389793976402884101, -0.00021093144903028035572374599, -0.00023045009102277296854366373,
-0.00007229013674218415913801800, -0.00003578305871748360640150280, -0.00023734955778653999964970811,
-0.00003561583216870238619965877, -0.00023049492413110690388136637, -0.00007206567010067558643263141,

-0.0002110304718965227434822329 3, -0.0001094548257981662302143824 3, -0.0001819443255913487952988433 2,
-0.0001468065055096514758686820 9, -0.0001470186956659013299743982 2, -0.0001817846054916146848745572 8,
-0.0001096928848533249765918526 4, -0.0002150205320531379409899897 0, -0.0001853875381105088079869425 1,
-0.0001115920996088387399935992 8, -0.0001496464044013666927143702 0, -0.0001497987938804795074107523 2,
-0.0001852698713089982632089280 9, -0.0001117616283369281585634291 7, -0.0002149448733316481631970551 5,
-0.0000736480306384209286061845 5, -0.0002348114949726790804535037 1, -0.0000364521462080368745050089 4,
-0.0002418304330965812291506789 0, -0.0000363235893945087213031779 1, -0.0002348469732821895777918896 9,
-0.0000734857007016584716030741 0, -0.0000721763745909064516554157 2, -0.0000357212313208313717136262 2,
-0.0002301319389389347772317123 7, -0.0002369975032134581747339519 0, -0.0000356337504856635867178139 8,
-0.0002301554146367136436182598 8, -0.0000720805141500003535449253 6, -0.0002107344128860830624692251 9,
-0.0001094401502896924028088165 8, -0.0001816994787428428232648608 4, -0.0001467338842867341619936177 9,
-0.0001468197436734343732994044270, -0.0001816303959144190951699449 1, -0.0001095353015829042124832867 8,
-0.0002106871475831943740526092 1, -0.0001710214372401646352496451 5, -0.0001381948502711113707045603 7,
-0.0001381832439130219159720974 8, -0.0001710086294984781648837596 2, -0.0001030982484423041722117891 9,
-0.0001983245950921147396708854 6, -0.0000679309619564844616125196 0, -0.0002165939481522992458156773 3,
-0.0000336181721734709240798251 5, -0.0002230388884276856957013246 4, -0.0000335731670255716083128838 5,
-0.0002166023846998158235849230 0, -0.0000679052145153201744268503 5, -0.0001983375749476203107717065 0,
-0.0001030835509166814307688286 9, -0.0000302212927613724981115023 2, -0.0000302235419158374793479129 9,
-0.0002005022423614099928712573 0, -0.0001947179923671681620912787 8, -0.0000611211945928302795511336 0,
-0.0001783121022944914231024571 3, -0.0000927684317912612105717223 8, -0.0001537658063308078304487480 6,
-0.0001243261871340634689905857 4, -0.0001242553096574600286772227 5, -0.0001538179298287123387099784 0,
-0.0000926968620441243106380935 2, -0.0001783402973946208417040104 6, -0.0000610741955648915157929199 1,
-0.0001947283373796509869067816 7, -0.0001056987251690965956595416 6, -0.0001307254272156473454136183 0,
-0.0001055363799722574213134362 9, -0.0000787298176988141749357494 3, -0.0001515125666933500646425120 3,
-0.0000518688619457179580390406 0, -0.0001653886820549496379072901 3, -0.0000256675722134768443999288 5,
-0.0001702671579400425426464860 1, -0.0000257071150885116235680887 7, -0.0001653536894502248306027675 2,
-0.0000519898339882711359866140 9, -0.0001514333485586396861165975 3, -0.0000788928378499531161903782 2,
-0.0001305975034620948053293298 9, -0.0001335034006622985442835210 8, -0.0001296427104874910141152355 4,
-0.0000202041576990349337652856 7, -0.0000408623660794125683273780 9, -0.0001187356937352174998450901 6,
-0.0000619911582370924252099672 6, -0.0001024061257397207815641321 2, -0.0000830196017473515586199833 7,
-0.0000827559570046525427812131 3, -0.0001026240543308132844614599 7, -0.0000617326684932345206993989 4,
-0.0001188804090637931063596452 5, -0.0000406680820770553345358706 8, -0.0001297116116179079196780482 6,
-0.0000201251750552886187113112 2, -0.0000567870960491941213386588 0, -0.0000423573648926948504367037 7,
-0.0000705961110059211160473735 5, -0.0000817008545544848399340767 7, -0.0000279018337021662385811354 7,
-0.0000890703754115873005091363 9, -0.0000138092416983661885672863 4, -0.0000916239455575586567289900 2,
-0.0000139238087683264939682431 2, -0.0000889552517800508467729625 5, -0.0000281662615922258787154019 3,
-0.0000814721791183001974772162 6, -0.0000427130918372246290480934 2, -0.0000702716235077951908430046 7,
-0.0000571615056859905220233014 6, -0.0000071072018685230503701232 4, -0.0000143887387563767801216142 0,
-0.0000448488068008139637664462 9, -0.0000410690390510249218642031 8, -0.0000217983492173407751915908 3,
-0.0000354246934009980221050480 1, -0.0000291166432970444047205211 4, -0.0000286254897526302296516514 3,
-0.0000358704645744766198282860 2, -0.0000213476550918627314460361 6, -0.0000413999313221732789433958 4,
-0.0000140605458705331124238946 4, -0.0000450228544515788036035897 1, -0.0000069624918963301015245007 3,
-0.0000462329109318980420234232 5, 0.0000002246791152700766570424 6, 0.0000004816749260629772964731 9,
0.0000004889951862731337200646 3, 0.0000003236346295915375323023 1, 0.0000007470666872733645095147 9,
0.0000001518610581051609513079 8, 0.0000009336993304956622479693 6, -0.0000000159752593147816587012 9,
0.0000009885006696978604219777 2, -0.0000000580925196801994238592 4, 0.0000009257700294972654188986 1,
-0.0000000482007876024236137980 4, 0.0000007979762332568738548201 9, 0.0000000414524326464591170526 9,
0.0000006473749121471378650583 0, 0.0000468003574401724506295723 0, 0.0000428996743710408385695694 0,
0.0000142764608362398672821271 3, 0.0000217049147002965689813667 3, 0.0000370033368593043442682823 1,
0.0000291983436763555583657125 8, 0.0000299049533087163946394604 3, 0.0000363113090879269703931987 7,
0.0000223121664798344134981143 7, 0.0000423458595361826214856053 3, 0.0000146972700569848194684322 7,
0.0000464921027829984223871812 0, 0.0000072602259965235724819740 9, 0.0000480725598978323919873052 7,
0.0000070777419371561030829323 8, 0.0000432810107392827349325273 3, 0.0000285069910641068368620156 7,
0.0000825950905687635123300638 4, 0.0000904650430981421022763600 0, 0.0000140891773378544216008224 6,
0.0000933793433827742564790411 2, 0.0000139017278579003019669223 2, 0.0000908280564864111189000829 4,
0.0000280656396013608100251969 5, 0.0000832370279694637297792161 9, 0.0000426287285375608106523411 0,
0.0000717957917198988956756014 5, 0.0000572400858211837649863404 4, 0.0000580192952582963224531474 6,
0.0000710120904380723180467430 1, 0.0001203833383428681429476100 9, 0.0001038330676406898497258352 0,

0.00006192240669891687482838394, 0.00008309469525665875077810596, 0.00008390578792607722500955514,
0.00010300121812758039553832579, 0.00006258715875818359006960145, 0.00011969106390390556713971409,
0.00004122096533532975401806220, 0.00013098303125910060315181815, 0.00002037599858503987373427313,
0.00013511635305782260002734063, 0.00002019407080782172699169597, 0.00013137857324053720670109513,
0.00004078093529010286559435133, 0.00005234954471178215960457808, 0.00002587883134899454044136613,
0.00016649349418400413627011636, 0.00017168185862034981243803844, 0.00002571317762172624241329491,
0.00016689317445149211391504374, 0.00005193466288569443666412601, 0.00015290857304592109277077694,
0.00007884675684891602855716014, 0.00013188042058524072496113189, 0.00010577246912910297216552785,
0.00010656630312335742445401660, 0.00013105468082139063346232277, 0.00007948646572211354763108637,
0.00015221385474413457556883134, 0.00009275223431934548432714616, 0.00012440355572374795184269203,
0.00015485901300309637775888538, 0.00012512830010899064921668644, 0.00015409710537769833295709820,
0.00009332802808153728359348567, 0.00017891747244875335883310119, 0.00006146429238704890509435719,
0.00019563624679433259164612613, 0.00003038600629161280648467096, 0.00020167483423115019087329403,
0.00003024702323752094363881228, 0.00019600983424611289154294858, 0.00006109853669347578390691478,
0.00017956355425588266033418150, 0.00021729386764661110095428476, 0.00022394623173360359274369769,
0.00003362128005204847520015141, 0.00003372421628176038894833880, 0.00021761287056678890782304703,
0.00006792007336981402225700027, 0.00019932597152122770358534387, 0.00010310376552645847273229074,
0.00017188644896625095018563578, 0.00013827097185156560221752642, 0.00013887814927907385104816074,
0.00017124231788059795437810517, 0.00010357934222891635389022374, 0.00019877640983226803924992232,
0.00006821455257841429445538062, 0.00018230925383158519165367750, 0.00014728808845052273408977872,
0.00014683821948869725635029859, 0.00018182656923916606147434427, 0.00010984671095598225567298994,
0.00021102242965774522750511955, 0.00007234097598022849436374959, 0.00023063008265034889185222944,
0.00003576519103914519034058406, 0.00023763783581554187422126001, 0.00003570603085499142020356120,
0.00023087139189595877854579986, 0.00007213616985492927720455963, 0.00021143688368655628919551281,
0.00010950145260021515328328207]]

6.10.1 Function Documentation

6.10.1.1 main()

```
def testCase.main ( )
```

Definition at line 62 of file testCase.py.

6.10.1.2 test_bFieldGeneratedByOptSF()

```
def testCase.test_bFieldGeneratedByOptSF ( )
```

Definition at line 30 of file testCase.py.

6.10.1.3 test_finalSF()

```
def testCase.test_finalSF ( )
```

Definition at line 27 of file testCase.py.

6.10.1.4 test_gaußLegendre()

```
def testCase.test_gaußLegendre ( )
```

Definition at line 38 of file testCase.py.

6.10.1.5 test_matElementsShouldGetValue()

```
def testCase.test_matElementsShouldGetValue ( )
```

Definition at line 46 of file testCase.py.

6.10.1.6 test_reducedSF()

```
def testCase.test_reducedSF ( )
```

Definition at line 33 of file testCase.py.

6.10.1.7 test_WeightsGauss()

```
def testCase.test_WeightsGauss ( )
```

Definition at line 41 of file testCase.py.

6.10.2 Variable Documentation

6.10.2.1 BField

```
testCase.BField
```

Definition at line 23 of file testCase.py.

6.10.2.2 bFieldCorrectValue

```
list testCase.bFieldCorrectValue = [[-0.000004669169876616231405156 57, -0.000004697747357037660250055961,
-0.000003959399954413595378499 48, -0.000004822210109380813629733 13, -0.000003965501952930266500805 66,
-0.000003877499659955074825048 74, -0.000003827719148226088557811 48, -0.000002928629677103681683004 70,
-0.000003099445883513666638440 40, -0.000003269003355319904813454 90, -0.000004549364004574882067474 29,
-0.000005587521011292382600835 12, -0.000004080591382310457715832 26, -0.000003551064005219094341423 26,
-0.000004350870218765218344788 00, -0.000006012692634775311500007 05, -0.000004220842037829208216888 86,
-0.000003181901388318364886953 79, -0.000004100099739415438254404 66, -0.000005969800942767407856467 26,
-0.000004375587461904082544402 34, -0.000002904547519473236613005 74, -0.000003827145101849383654010 23,
-0.000005488115761224373012573 33, -0.000004525002253112012802190 12, -0.000002896738240265727740277 22,
-0.000003557979077100258930130 46, -0.000004729170963651482885209 58, -0.000004642092668103783633802 75,
-0.000003259278409432724355664 30, -0.000003311429737662627467833 85, -0.000003918188492096576364673 91,
-0.000002099634375445946809276 43, -0.000003348456060496892882554 03, -0.000004422316650016540052993 07,
-0.000003407782150417815574265 82, -0.000004964014997046636271018 50, -0.000003710205891880322134706 27,
-0.000006471742346850464589543 43, -0.000003101840589612448290622 85, -0.000007416109602733633789106 19,
-0.000002302115862790705356897 46, -0.000007542180563880607090034 45, -0.000001661047892629223233262 64,
-0.000006880556359289997411636 27, -0.000001524211251690854259350 86, -0.000005707651642471200325549 15,
-0.000008342183894649755178954 76, -0.000008575096782296751047941 15, -0.000001716981418177732680188 20,
-0.000000833627055611081694332 66, -0.000007791492667078616981960 78, -0.000000611222253944708138039 95,
-0.000006344593299394096200163 19, -0.000001329859367882307268551 45, -0.000004747682704468811800259 47,
-0.000002945599455494238689640 08, -0.000003495155480302092057206 83, -0.000005061521488488576830347 84,
-0.000003631924834526299678905 47, -0.000007058926325864303918901 02, -0.000002803166375796800723204 60,
-0.000004843725451266400324422 91, -0.000003517594473839403047436 35, -0.000002784455554348361045819 34,
-0.000005067907102278958375227 42, -0.000003605431344164336267249 91, -0.000007230023054888917250155 11,
-0.000002701544595534071612722 31, -0.000008627252206377690322463 07, -0.000001516034256315816376996 68,
-0.000008896818454346353802122 27, -0.000000545749437405359894701 02, -0.000008074595444233454712926 88,
-0.000000287684156485875496321 95, -0.000006539798017677640283443 81, -0.000001048520215178691083511 37,
-0.000001729847239258692423514 76, -0.000000839834326124461531782 54, -0.000008439008115890682154743 02,
-0.000007675821183079086331476 49, -0.000000597268567944310013124 51, -0.000006258216888525444950128 37,
-0.000001284687519605434690500 485, -0.000004693463019647930058576 93, -0.000002865153004397145685826 78,
-0.000003470761878876645504734 97, -0.000004949419569968011847413 41, -0.000003634935900347052473181 36,
-0.000006924696060383603073097 50, -0.000002812678644541278844166 95, -0.000008199362443059459207815 59,
-0.000003171962524546940482986 57, -0.000004713747175559267898895 32, -0.000003715651603631717748402 36,
-0.000003360228058128201781591 80, -0.000006173589524796339052627 79, -0.000003119009788903953190816 48,
-0.000007106838088744919283376 06, -0.000002325287924573052330488 30, -0.000007257567319432155692989 55,
-0.000001671477311721885492592 23, -0.000006646439844438127494139 28, -0.000001494427110945597850732 12,
-0.000005536930259193586047802 15, -0.000002003327756573957827548 54, -0.000004316458287959380088992 78,
-0.000005529433366181379469364 40, -0.000005137848765220242051648 61, -0.000002913492218500592313208 02,
-0.000002846558478894211337349 54, -0.000004480448295283521281098 72, -0.000003105741629932859720935 25,
-0.000003766094683489923555699 39, -0.000003674135521447355052886 14, -0.000003200687551633199015701 85,
-0.000004412662744884351176931 10, -0.000003834529390517351181579 73, -0.000005101301723723616199498 64,
-0.000003572254233953652774409 33, -0.000005518687668101426204293 57, -0.000003209646821722285806244 77,
-0.000003013788779416146341538 86, -0.000004107786858267637268018 42, -0.000003886649395090982258128 92,
-0.000003972274445804966337366 69, -0.000004100691593327836342842 11, -0.000003686432516574610400249 66,
-0.000004243769233644540846952 99, -0.000003520000496158839291098 44, -0.000004371178943849891816437 85,
-0.000003376627843257072685126 44, -0.000004443780700306835564072 92, -0.000003244387453385726021207 97,
-0.000004425754754760034221446 23, -0.000003121464441778340444456 26, -0.000004305061406002007946998 42,
-0.000001625397001290687052922 32, -0.000002010421170774849839644 98, -0.000006025176983159446724813 53,
-0.000005740816591202888841940 97, -0.000002475171262408580079907 59, -0.000004948165877237167161715 57,
-0.000002824910051833316077289 03, -0.000003831068787161905272455 83, -0.000004106338473534577906126 596,
-0.000002707293732265527070896 68, -0.000004620230251661191355090 89, -0.000001885700362389204486434 73,
-0.000005263276692959339155245 33, -0.000001531639762121057001811 27, -0.000005810663830738314987727 37,
-0.000004214427283803097153223 19, -0.000005047567038276975362530 27, -0.000001709896970175021093158 35,
-0.000000368205651246759689274 25, -0.000006104964874039769308507 71, 0.000000147970715399805712407 52,
-0.000006998154097385826055103 14, -0.000000139543256453925005840 17, -0.000007325311268714011280823 22,
-0.000000957449086117795926051 14, -0.000006814847503880925708760 32, -0.000001919098645207326594053 39,
-0.000005462538042366831693267 84, -0.000002659382888561019648448 76, -0.000003582988164699999556671 26,
```

-0.000008127960838667446259319 00, -0.000007457537766348704991409 37, -0.000000234238167987601168211 21,
-0.000001530497702892756555463 50, -0.000005735090100133847753390 71, -0.000002538748528520541344422 14,
-0.000003362471878848714567928 75, -0.000004277978857843769559853 38, -0.000001009160251116196961387 07,
-0.000005313268141089583862588 92, 0.000000665965933245403488290 42, -0.000006633416101256644790214 35,
0.000001286449913954700983257 92, -0.000007741212825529410768436 90, 0.000000871395950407652559332 63,
-0.000000686976806836211766085 19, 0.000001087106470068485688851 07, -0.000005372574302659216289191 49,
-0.000006762387901441110924850 07, 0.000001735673372247141482926 88, -0.000007920443808640041069872 24,
0.000001272895962333544302592 24, -0.000008304913740411444737061 18, 0.000000062266992184529590216 48,
-0.000007564410539416802209118 47, -0.000001360573441233412476806 04, -0.000005713940709338972224811 89,
-0.000002477620552432487205092 41, -0.000003185525822041595355317 13, -0.000004285070874239019008832 17,
-0.000000099761910657629347219 19, -0.000001427228406237246719401 23, -0.000007121890788624390749520 65,
-0.000005403874044001729978867 37, -0.000002481624482490750735133 46, -0.000003073674819372230883302 93,
-0.000004232310602584853629661 02, -0.000000777199925254534555820 50, -0.000005212217344728846509167 07,
0.000000853268705209387845752 14, -0.000006467267976458186151970 85, 0.000001450366431284919655866 67,
-0.000007504937291090763591802 55, 0.000001022957800677181251607 08, -0.000007828939574459940902730 58,
-0.000004852333446753194925538 20, -0.000005787573750391326826518 75, 0.000000020287080273935243494 07,
0.000000494123708124895733633 46, -0.000006552513892308689661726 51, 0.000000176526386444774723423 77,
-0.000006767676752540858455193 30, -0.000000683554916541925163217 55, -0.000006192424092432093462169 51,
-0.000001712915954191518598656 24, -0.000004844962781809541334590 57, -0.000002546711824114191282178 44,
-0.000003032326826771812897331 99, -0.000004125421682941469509386 60, -0.000001249268188050931304157 69,
-0.000004900055231256016360240 24, -0.000004100792051189442145528 19, -0.000002167802257612540191332 80,
-0.000002659935093759879472431 36, -0.000003044511846408271801877 12, -0.000003978446178241522450977 18,
-0.000002008865214462643643099 23, -0.000004343371960747792585877 13, -0.000001264973063471340136853 40,
-0.000004820345137294431170119 19, -0.000000970477087679776405311 54, -0.000005199108208945110929364 61,
-0.000001123954920169712695175 01, -0.000005269881142691022743776 95, -0.000001590297779107672852580 53,
-0.000002790076419657463466597 34, -0.000002707920534783716912688 40, -0.000003705861395300735868418 76,
-0.000003639831032048499611622 66, -0.000002670940549635263423431 54, -0.000003547311581061626330049 58,
-0.000002674698768215354493561 50, -0.000003419714389657307361184 00, -0.000002717673792120109748461 43,
-0.000003259265982187504637304 95, -0.000002801549574795857155024 32, -0.000003083134575019942040502 83,
-0.000003811687890361988188879 34, -0.000002918682155814867151366 44, -0.000003758574669448968346373 82,
-0.000003275371435838905081349 87, -0.000002948112853648078168658 60, -0.000002431318723147511385679 30,
-0.000003126679961502833175648 45, -0.000003648708365475053834204 32, -0.000003827816526512422955328 45,
-0.000003183129155410189311398 16, -0.000004313944058749856084794 20, -0.000002607368642943563552788 04,
-0.000004447290998259228003187 95, -0.000002102292897194702951088 65, -0.000004223755108950629635965 02,
-0.000001848759477094594787843 04, -0.000003767219061504040205539 19, -0.000001960451867220521750473 98,
-0.000005923420840851475748943 45, -0.000005544760707924370150411 36, -0.000000814003347248115565701 91,
-0.000004253244226850051261108 3, -0.000004699591704416164490286 81, -0.000000737345699662157190757 71,
-0.000003754035505591927710645 04, -0.000001737684900976831395982 64, -0.000003076109720272050648728 47,
-0.000003165602208531349471594 28, -0.000003512825164681814800536 19, -0.000004596921397320425354713 83,
-0.000002701527771238801390441 38, -0.000005604500464185683460084 31, -0.000001687199993214851172440 79,
-0.000003423634737932706773077 05, -0.000003165555164106575179364 50, -0.000003199467932657861860961 40,
-0.000005114769646552445833177 52, -0.000002384952630150790012382 97, -0.000006470135878608972665752 74,
-0.000001082573787397974524103 57, -0.000006914685681239927260075 24, 0.000000031670643635583581195 60,
-0.000006434154519761684247382 98, 0.000000507715209298222229401 54, -0.000005329878432634424523974 09,
0.000000062410153394725731244 98, -0.000004080209874444685043360 86, -0.000001287619385107277441640 81,
0.000000305300963399226811309 37, 0.000000804931633721443997277 14, -0.000006759106531276178780106 44,
-0.000005563573402434483275143 34, 0.000000310536619466591689908 48, -0.000004204891421702167735529 29,
-0.000001158304945279706829906 53, -0.000003203034977046413815985 30, -0.000003231981014917997459143 46,
-0.000003394080338639974761408 63, -0.000005310010027204825310540 89, -0.000002280563636248469533586 57,
-0.000006784693242381760576659 77, -0.000000884776552097538224539 25, -0.000007274133226560342348231 74,
-0.000003268978054318318354865 329, -0.000005161678927616114361952 47, -0.000003428506799238336935137 98,
-0.000002404325456694791921576 65, -0.000006506812171696135195986 17, -0.000001125054147143240300041 59,
-0.000006951900538091945776892 31, -0.000000038109380746886272567 05, -0.000006473838782693360876770 51,
0.000000414566471161723996374 99, -0.000005367828254428663770656 07, -0.000000040386984611623721950 87,
-0.000004110903410176831024986 33, -0.000001380732056537456376377 15, -0.000003183728073491490218310 11,
-0.000005627275287447657896560 80, -0.000004776581892182495043305 39, -0.000000612278458906465059219 61,
-0.000000948270888847624863399 24, -0.000003815478480871763777089 63, -0.000001934438735728849900871 64,
-0.000003112106362083816687080 44, -0.000003318697971439304780631 21, -0.000003521997951986014315948 69,

-0.000004704946597670969516812 73, -0.000002738495074569903078875 49, -0.000005688898621621768605146 74,
-0.000001769528991031094350896 57, -0.000006004517856198477860068 13, -0.000000951497156201585765080 38,
-0.000003661100896278850712182 60, -0.000003234185591876674678596 00, -0.000004026008636874376506436 38,
-0.000004469495847771409014577 08, -0.000002724001468378236545539 67, -0.000004586721590835389027304 62,
-0.000002302351486545032412730 11, -0.000004355930598681501865628 54, -0.000002128833052767957807244 646,
-0.000003885539920361719780144 90, -0.000002287528065415136106905 07, -0.000003367620965455409258058 81,
-0.000002750529684441294707799 61, -0.000003001205317270282573659 20, -0.000003390656970543113683374 27,
-0.000003914326068653997588988 81, -0.000003867189823242919772985 74, -0.000002837600839175084593070 69,
-0.000002840604460082159842673 5, -0.000003720199167780288467612 42, -0.000002870573199346581707468 64,
-0.000003490914196441135331953 16, -0.000003825832351644726756509 46, -0.000003245016470337282255470 34,
-0.000003818641803553966068288 62, -0.000003048967944455735619121 21, -0.000003847908125245524118163 58,
-0.000002926265125261060909717 68, -0.000003892308380863364904416 87, -0.000002862738197587269642238 56,
-0.000002492694581753588705861 06, -0.000001655482237209619174163 01, -0.000004406155336826438219778 94,
-0.000004975340784626303956589 71, -0.000001287683300810760832579 88, -0.000005486593961727192739652 89,
-0.000001385756011294490192730 30, -0.000005706228818302739558515 45, -0.000001801015181645569025002 26,
-0.000005457598964741178506026 97, -0.000002320718332930050728204 12, -0.000004707444945448636833360 69,
-0.000002743208692611579841867 32, -0.000003616020659077754512405 33, -0.000003992667533797232255841 22,
-0.000000943664169711307707902 31, -0.000001893959634366278747664 54, -0.000006824526704582180925199 01,
-0.000005569244589370603003308 27, -0.000002641951482764336413701 443, -0.000003753790329346908617768 12,
-0.000004138060931600643000847 70, -0.000001891613888080865604464 45, -0.000004910989905867787144321 15,
-0.000000511858972162371407152 39, -0.000005940090633514393263396 37, 0.000000066869698384312723880 37,
-0.000006849347666819941143358 76, -0.000000162249097066521752707 46, -0.000007239504929639869315930 42,
0.000000191391818799561805343 25, 0.000000914529967001440018990 81, -0.000006601026404545618444402 21,
-0.000007780046901856716898802 92, 0.000000606924975258649377351 26, -0.000008288517969195865596641 12,
-0.000000407243207521755645005 55, -0.000007768517864930579020074 29, -0.000001632687759934762493114 08,
-0.000006184252258447503232798 11, -0.000002585877388757879148791 99, -0.000003894962601007848762828 95,
-0.000004241965490617003539544 71, -0.000001547524222414936022091 27, -0.000005260361892204286456337 70,
-0.000001584287326956026389354 14, -0.000002587251259447109409157 64, -0.000006477356464652411522243 26,
-0.000004041447503927193620506 91, -0.000004290813773417610484193 36, -0.000001538564664527889685189 31,
-0.000005405471948075501606651 49, 0.000000322772421190406936908 86, -0.000006863814565985649504911 44,
0.000001105656232090204249795 79, -0.000008145016782084720281136 88, 0.000000788619495527599086132 74,
-0.000008703714485079735570737 49, -0.000000285107883780583305326 62, -0.000008159446728336942360648 89,
-0.000006695377111149468838303 916, -0.000007898137635483467843271 37, 0.000000591116908220294016904 7717,
0.000000339588975320378274061 43, -0.000008434503701231834684685 22, -0.000000607310206707123813079 03,
-0.000007954749258240881776066 06, -0.000001763608810331004089839 62, -0.000006426076931872222163444 007,
-0.000002649509538613160246274 84, -0.000004197271373554537065978 83, -0.000004279506244418695642766 32,
-0.000001893290769712060806547 35, -0.000005328855800032962806787 30, -0.000000162181045123928933213 14,
-0.000006056476665462023338158 07, -0.000004357756855900262932582 67, -0.000004212102411765054810926 29,
-0.000002766586451897782492737 69, -0.000002576634559143457449306 28, -0.000005046894628565875934539 47,
-0.000001207757487523629997183 60, -0.000006128723680501494967451 90, -0.000000567284229447634383124 02,
-0.000007087422650970049133544 41, -0.000000685820997888661863438 56, -0.000007535507417486027727064 61,
-0.000001335510027280870001933 95, -0.000007201997199238423191618 42, -0.000002150389688223353754537 20,
-0.000002208641245363017065433 78, -0.000002144558714932919257084 58, -0.000005846995424213146957286 62,
-0.000006157734472189378818534 83, -0.000002368629268134221628152 50, -0.000006032276338722278724309 00,
-0.000002692053956048653862631 52, -0.000005441910208662434093710 56, -0.000002923694335091058528984 78,
-0.000004515210563754727400337 56, -0.000004101112533974774877287 93, -0.000003501150522541799881774 25,
-0.000004607095381496415074809 67, -0.000002671269673319149039756 11, -0.000005257454871785676784440 82],[0.↩
000002366897358546120360344 93, 0.000054013542118938742209524 61, 0.000053612097522474058588614 48,
0.000002126343868342701241505 20, 0.000213763439098918608732499 69, 0.000216019333245713852613754 946,
0.000213713517176211045026171 111, -0.000213003594260546146170305 4113, -0.000210615828168400626282594 22,
-0.000210617417978744331561309 140, -0.000049348730125814980118290 07, -0.000049411245329527477151108 92,
0.000206397270217370624187727 68, 0.000206239154234050270104311 45, -0.000096773531865318148173504 83,
-0.000096692041844518845754297 14, 0.000192744965932244789914851 16, 0.000192447502414842309698822 52,
-0.000136642271729642527646961 25, -0.000136485020894412602439349 50, 0.000171294275122681551586498 68,
0.000170866876920469819289838 40, -0.000167401237717685209263271 40, -0.000167241831956774895072415 09,
0.000140841761425895129053106 89, 0.000140337208521422820463320 22, -0.000189147258642013077001645 12,
-0.000189037851693255715655739 29, 0.000101246844344530625612287 03, 0.000100749101690680243552204 75,
-0.000203053568816226320944626 22, -0.000203010047424189462965393 73, 0.000099642881706784967995801 93,

0.000052831829908935487821568 24, -0.000202762728804438017646272 21, -0.000210563229658998588315552 14,
0.000001835674632618827812068 12, 0.000213603064385481356511620 19, -0.000049192200429969061927816 53,
0.000205862114994704177009174 35, -0.000096089662611685525855399 76, 0.000191725693585311930781506 305,
-0.000135716079690289116548018 72, 0.000169835367989672062286055 09, -0.000166529004595069744570565 16,
0.000139149903398607174961551 51, -0.000188535032811145954407053 28, -0.000095091484550649459668489 46,
-0.000134477670238808877292008 28, 0.000190682240417582883878350 19, 0.000168345792803651343348333 99,
-0.000165388688111907226387098 38, 0.000137443584065138923490378 97, -0.000187726715099097597044819 35,
0.000098070980947417005831739 93, -0.000202356178904872983882398 67, 0.000051756810241653146160284 44,
-0.000210464845837929803891441 78, 0.000001498921884865296601756 35, 0.000213446731543043581020162 75,
-0.000048764385344397459596914 31, 0.000205319072742291577849879 65, -0.000201858604196821200251962 81,
-0.000210340371938423306447801 81, 0.000050531091609622903900421 45, 0.000001141549011515240626328 07,
0.000213265954746442304434950 19, -0.000048221469450230348703211 74, 0.000204687954381726046684808 47,
-0.000093875703566336882134255 04, 0.000189469198724572412653962 3, -0.000132979050053057051766006 39 38,
0.000166616599924370020410768 77, -0.000164009304483600152181968 50, 0.000135467948704497647669913 7096,
-0.000186745968858859631737752 71, 0.000096259069368213214640782 33, 0.000188264518893218539118822 32,
0.000164904304867192097894951 22, -0.000131458842833363553958062 05, -0.000162610429829441994164674 59,
0.000133519280592369536620964 14, -0.000185749833658335994053881 21, 0.000094481658016206203610333 04196,
-0.000201351175273233376755016 58, 0.000049341243747957562069144 7687, -0.000210211566384638595155145 60,
0.000000815790807157689890866 48, 0.000213085783461415003391115 08, -0.000047652405906943999286569 38,
0.000204059910509638014900271 61, -0.000092636945549539746985572 01, 0.000048374668358535115688579 20,
0.000000581557010391455304552 90, 0.000212931160209205654716344 00, -0.000210100436600453799535284 66,
-0.000047134694282584487206710 23, 0.000203525490709004219791461 99, -0.000091559767661243121258940 47,
0.000187244227575668106320760 870, -0.000130149913836981865646047 48, 0.000163461342399836130710039 09,
-0.000161409759720115158782710 60, 0.000131888131489667497116269 33, -0.000184895451859136612272893 62,
0.000093008792768913116639174 628, -0.000200915576365324875405746 21, -0.000129252705394381390028787 91,
-0.000160593503341628087658504 72, 0.000162495428603707777267532 55, 0.000130808091601162553149415 02,
-0.000184317129874974767529015 483, 0.000092050348821694324374186 3, -0.000200621656580162882861266 04121,
0.000047768624093589003389882 80, -0.000210025819094112627507045 52, 0.000000470923799633684671666 07,
0.000212823222202876546557681 61, -0.000046746451295604886171878 44, 0.000203161017136980489401418 27,
-0.000090807110697663153287689 80, 0.000186555387130252393760102 00, -0.000210000457390162076657877 39,
0.000000468496722975468725434 71, -0.000046570894974512044676984 16, 0.000212776183591962212649886 88,
0.000203017284636241576245910 27, -0.000090508659689697462430620 08, 0.000186294559891074982750080 11,
-0.000128912599494081291182784 32, 0.000162139397234438431549818 92, -0.000160291530754248264098102 42,
0.000130421087905862660056092 74, -0.000184106934719242398140748 31, 0.000091721448271931169766643 67,
-0.000200516910719880100217801 80, 0.000047580422904333251820589 02, -0.000160557719816197100358781 68,
-0.000184300847421255760590075 30, 0.000130766200997182237579161 13, 0.000092041590851325123193854 56,
-0.000200619233900458353173426 60, 0.000047799011715865603639082 08, -0.000210029014512176810929727 78,
0.000000532571303976980690958 70, 0.000212795264389177541351771 96, -0.000046670850250016787761232 07,
0.000203112219705347667739539 50, -0.000090735178138577786606001 90, 0.000186494734899948858821913 67,
-0.000129196355327063495781192 43, 0.000162435842229606636980605 80, 0.000212875970359699450210055 89,
0.000203428421306936594312522 74, -0.000047051586169250888159066 44, -0.000091470201312729388721037 28,
0.000187121525124687927872646 49, -0.000130073492317495011810768 45, 0.000163334936925583942401302 15,
-0.000161358668415655109743336 86, 0.000131784712822059215872605 44, -0.000184872418188891028635253 98,
0.000092952643557140575696247 36, -0.000200913995275991171928636 76, 0.000048376858798901613983606 96,
-0.000210107320454922511963544 16, 0.000000634559435855381146126 54, 0.000133329104367413051306784 37,
0.000094329593865078435093243 78, -0.000185735537609260761881049 83, -0.000201355965797107107855233 885 61,
0.000049242235424952767071371 53, -0.000210222976609329898801947 77, 0.000000771610574274012064539 59,
0.000213004828755085086945447 91632, -0.000047650083777419015805803 39, 0.000203915828729739876962884 23,
-0.000092603441979978340001765 37, 0.000188078223405378422558240 08, -0.000131414566344109873203908 75,
0.000164701630295302340646623 75, -0.000162575441295885285680061 78, -0.000048358240542781555251859 03,
-0.000093952712733040200554675 28, 0.000204499160760300102040282 12, 0.000189219154298339451091298 30,
-0.000133010853923613869050773 81, 0.000166331494185268286046457 17, -0.000164020307842087641756259 61,
0.000135174630940144074397398 37078, -0.000186756679712913014734104 73, 0.000095981201274371452246586 93,
-0.000201875961792950960570214 30, 0.000050287066748564196905398 05, -0.000210357235307407665345856 35,
0.000000945831546361488028777 70, 0.000213161468552725145386119 009, -0.000187775075555341541841392 347,
-0.000202391314814704703962609 09095, 0.000097661762708549853606934 68, 0.000051361120933515946256631 25,
-0.000210487880383046306764119 57, 0.000001142268415573614673906 69, 0.000213321737595180828221674 84,
-0.000049055358650310537703756 25646, 0.000205089184217387613955995 05 46, -0.000095300402920074772108137 43,

0.0001903705556104428621693597 2, -0.0001346093178988341267348499 7, 0.0001679774430417616513672857 1,
-0.0001654658189289348396103923 6, 0.0001370433524960852948119255 4, 0.0002055962732426158451250891 6,
0.0001913568805607198286718173 6, -0.0000964293457346822378666029 2, -0.0001359533238758213892060505 7,
0.0001693870992367652722736942 6, -0.0001666806139842859357802495 9, 0.0001386463182725114820764006 4,
-0.0001886276331862692264527969 8, 0.0000991087138051884193861579 3, -0.0002028186162218053624484159 8,
0.0000522950078601460259801643 5, -0.0002105926765739640684621525 3, 0.0000013303590015119418810574 5,
0.0002134613894022320795384323 2, -0.0000496286679993075273593927 7, 0.0001000956685114009499628154 3,
0.0000529420458064071519600421 5, -0.0002030867487635261707697165 2, -0.0002106528585584195643973931 5,
0.0000014827972310789902995738 0, 0.0002135597787990394381529152 3, -0.0000499818687731115373967458 2,
0.0002059442087401272172124692203 1, -0.0000971529286389914904136302 9, 0.0001920284777011037537563674428 2,
-0.0001368220246578469301290681 4, 0.0001703444968695807697574059 20, -0.0001674657703063968448679854 6,
0.0001397352856229101597442671 1, -0.0001891745623195150230455574 6, -0.0000973408199370431745844553 5,
-0.0001370641656027741703792466 3, 0.0001922855098764846405821199 4, 0.0001707063404805534616390932 9,
-0.0001676862743764999884830968 5, 0.0001401473610840748022701024 4, -0.0001893215969887804034539613 4,
0.0001004772985978321362711407 3, -0.0002031481519257887630430098 3, 0.0000532127396320849872725540 5,
-0.0002106561186949413644938639 0, 0.0000015904738675548165585113 1, 0.0002136030003239068706267361 8,
-0.0000500428453328594430500088 4, 0.0002060819870101862768489026 1, -0.0002029865128635665000138288 4,
-0.0002105986254693183656536148 4, 0.0000530872648018523764917864 2, 0.0000016619693053623359908224 0,
0.0002135859791265856590797289 6, -0.0000497766111758949335048402 0, 0.0002059916874937127977131853 45,
-0.0000969407501994995441730110 2, 0.0001920936933358060910888298 6, -0.0001366230101830284346647176 3,
0.0001704249089010146387193622 7, -0.0001672931292119931819591810 4, 0.0001398309590068383726456369 1,
-0.0001890353326868501773576347 6, 0.0001002116858937327945836687 2, -0.0001355482223529339545244065 7,
-0.0001663324566789901079021801 7, 0.0001695543057630174274297496 3, 0.0001388509117055680155104757 7,
-0.0001883490989604627444467532 8, 0.0000993613632285822890189558 3, -0.0002026196219942150774252082 0,
0.0000526106411672741937184116 8, -0.0002104857493531503604983906 6, 0.0000017133739089902382508005 9,
0.0002135131937119784514486631 2, -0.0000491985024210138695136863 8, 0.0002056911169589447870888199 2,
-0.0000959920459015723228930056 0, 0.0001914892460023451313620018 7, 0.0002133979156093022879620013 7,
-0.0002103313751620800747289818 9, 0.0000017596611473298941870428 4, -0.0000483772188227479612199384 6,
0.0002052308499640048981751266 9, -0.0000946232270021666487707159 8, 0.0001905726008140005110769177 3,
-0.0001339900448056123008142015 4, 0.0001682403092978662921715138 4, -0.0001649391631134404664037196 3,
0.0001373751132188857557430566 9, -0.0001873582375785165835511969 5, 0.0000980788441487153708417401 4,
-0.0002020969105099591203375538 8, 0.0000518799675970007214049518 9, 0.0001666962668297994169547965 3,
0.0001356460018845133874351716 1, -0.0001633155458414857572752021 4, -0.0001862054124214689617821533 2,
0.0000965801982299588829055866 9, -0.0002014920377858509740586628 9, 0.0000510275960346351042354502 7,
-0.0002101559035215930292071068 2, 0.0000018125145185340493495825 1, 0.0002132600923426924402395016 3,
-0.0000474224418152625377823397 6, 0.0002046861900128961356086498 8, -0.0000930299879713407309057143 8,
0.0001894921865101558246594071 2, -0.0001321746048184879751032083 7, 0.0000018818309673382172091804 6,
-0.0000464630617822779819914254 4, 0.0002131232018709166154812190 1, 0.0002041453653590416636616489 1,
-0.0000914405985686857326799291 80, 0.0001884210742155565830244223 8, -0.0001303663826355080002268593 8,
0.0001651693277844899028408004 5, -0.0001616991041747481037613165 5, 0.0001339420393183044302731765 6,
-0.0001850585356953765017382074 2, 0.0000951109878854195883680564 0, -0.0002008917251981771423877121 1,
0.0000502016180224585973094694 7, -0.0002099832446218229456674081 4, -0.0001603259748784645157069290 14,
-0.0001840850692673668906069323 1, 0.0001325353350988635922191177 8, 0.0000939097041189565387257765 0,
-0.0002003826151704963100339679 2, 0.0000495447767721597897823751 5, -0.0002098372668729149787339438 6,
0.0000019757882177263054767669 7, 0.0002130105710721140173467130 8, -0.0000456262778097446842508720 7,
0.0002036958107073732159291584 7, -0.0000900802204393742517347931 6, 0.0001875301417385050900023257 8,
-0.0001288272748706958709553199 7, 0.0001639023661667604635835265 9, 0.0002129417252995247241374732 2,
0.0002034106510989128278454191 0, -0.0000450216600430950172136331 0, -0.0000891398897212229403492003 7,
0.0001869618185935924075356806 6, -0.0001277787168388340552534210 3, 0.0001630969461315070416062428 2,
-0.0001593957703245123226024754 2, 0.0001316505978162389506238433 3, -0.0001834267777761446482836389 8,
0.0000931719701627960525782204 4, -0.0002000382044621549278845540 3, 0.0000491723512988372925790192 5,
-0.0002097382445859562822303382 26, 0.0000020974738069065830590000 7, 0.0001314305911971278809004942 7,
0.0000930195972104949224705181 0, -0.0001831785847507890092888960 3, -0.0001999078362820253819930049 8,
0.0000491520639305229235403380 0, -0.0002096998442457208983169536 5, 0.0000022405772381956269206145 4,
0.0002129293297837518954968016 9, -0.0000447286526541879731443118 7, 0.0002033374488485800988989421 24,
-0.0000887508172796281448240982 0, 0.0001868083179952192182226439 2, -0.0001273700605665939445104023 5,
0.0001628825697361331798548173 0, -0.0001590422259271423639261089 4, -0.0000447846148777276625656783 3,
-0.0000889644190914527956256727 1, 0.0002034909106058223728433209 7, 0.0001870976861001771611300922 58,

-0.000127656118197188898720804612, 0.000163297116830722741000897 49, -0.000159313367280194387223118 96,
0.000131914332645102636487263 74, -0.000183374562143678540050706 35, 0.000093481555099488952589696 64,
-0.000200009391886011775579107 26, 0.000049493066252138007577593 41, -0.000209727106234931624665143 72,
0.000002389830507354445376087 29, 0.000212977189160388894785536 08, -0.000183983157634391570484478 9,
-0.000200326754064990940942639 02683, 0.000094492160348649562734424 38, 0.000050148708940738899496672 44,
-0.000209815721325538521004408 50, 0.000002528651248754036990137 74, 0.000213079607675264759989244 84,
-0.000045174213213229682942814 05, 0.000203850633158514978754416 45, -0.000089741542122590166525621 10,
0.000187789767597601014699829 07, -0.000128588032791025546827667 90, 0.000164281792611773287899126 77,
-0.000160164523721893103285277 00, 0.000133032596131738614078296 46, -0.000090958732543672171015451 68,
-0.000130021814633640676265452 51, 0.000188782504689455967209973 93, 0.000165690802020247639280814 317,
-0.000161465995179999673332649 21, 0.000134620033002713147037704 42, -0.000184912395731896654627046 67,
0.000095903597144704472296158 90, -0.000200812383100140587561047 16, 0.000051026767893961046357147 18,
-0.000209952706433970848632492 02, 0.000002644524928086319488193 84, 0.000213222204449962763912057 58,
-0.000045830319339305121330990 78, 0.000204364180973693375256086 35, -0.000201394570547478162381854 76,
-0.000210118371684065545768898 38, 0.000051998243801403415127303 93, 0.000002722801468218608437793 68,
0.000213384057868348597048965 83, -0.000046652954416739657328167 67, 0.000204954985058026145422135 91,
-0.000092433521298737074726147 02, 0.000189927370851195238300734 85, -0.000131743506930348693289428 71,
0.000167313375633493006755317 83, -0.000163023823792382143186771 58, 0.000136439678741642723362043 07,
-0.000186024069735272941936501 94, 0.000097506112361717419617460 576, 0.000191051566394959875059417 21,
0.000168904900873345534088876 27, -0.000133502055240911966273176 24, -0.000164608907983626266877519 1298,
0.000138217074869828728889489 81, -0.000187154488468579011670525 62, 0.000099056341249655630379344 91,
-0.000201988248481109147107817 60, 0.000052911332290750669681521 36, -0.000210289276935236956958058 36,
0.000002742154550548674764760 79, 0.000213540855133185147277791 18, -0.000047530009965808140835925 87,
0.000205533858611026284089751 13, -0.000093954786616974972393951 25814, 0.000053619251863530922430200 33,
0.000002686323381411473377664 01, 0.000213668570531652684705778 42, -0.000210441731188790139839463 68,
-0.000048344223792715597457452 61139, 0.000206012424509344443772362 88, -0.000095306526554942740114828 58,
0.000191983803473363018885292 09, -0.000135042611600624649524290 95, 0.000170223203178939461409999 56,
-0.000165989795212327477524905 11, 0.000139681354849841357634865 35, -0.000188138727297914161397038 30,
0.000100315525242880511282844 71, -0.000202507742391278895048925 55, -0.000136141432890715322187244 47,
-0.000166966415908708623507678 00, 0.000171064615396395857272598 96, 0.000140606082455802211365891 72,
-0.000188835219833081524674206 49, 0.000101087880727787415369514 63, -0.000202879598830685795796808 16257,
0.000054008953000139444181049 16, -0.000210555305326539922384718 18, 0.000002556753483283197157560 67,
0.000213747120039270405806794 42, -0.000048980454669934835021903 60, 0.000206316471213195830975914 44,
-0.000096293140202926750213785 48, 0.000192579974649746666313931 96],[0.000242214224195631957570543 08,
0.000235121137550970939677705 27, 0.000230580578631971064252834 11, 0.000237487591631565554684993 94,
0.000036420799359195195664001 05013, 0.000000048761546279976064480 55, 0.000035737720573537822758673 77,
-0.000000020882863264937369152 15, 0.000036430617731940654507091 09, 0.000035695179119540193773268 97,
0.000235270441830711844006726 60, 0.000230633325132287007005055 85056, 0.000073583378879772868663526 61,
0.000072204463049522785915261 55, 0.000215429096872345350729624 49, 0.000211146302438801840685791 98,
0.000111696296081046123841590 27, 0.000109600347963228198682264 50, 0.000185727368552101169789370 07,
0.000182009695976221354827076 50, 0.000149770571352532555752520 285, 0.000146949218144508321915595 26,
0.000150036034171210176467627 02, 0.000147017942870696224303556 09, 0.000185434879755141051411657 10,
0.000181919406624720200347719 58, 0.000111890192767867863020208 49917, 0.000109632750768500912596017 94,
0.000215174299851819289287926 13293, 0.000211060245571116937221556 32, 0.000073685463939605411794880 88,
0.000072197223498588641334154 15048, 0.000198827411334484434050790 733, 0.000217171399645370939626015 66,
0.000067934319007293702164136 66, 0.000033587345328891007043611 16, 0.000223628993371385333382289 63,
0.000033682563676600931566675 175, 0.000217131046975615338160206 51, 0.000068050531687914142032515 27,
0.000198749821646811072059848 89, 0.000103290237682533833293851 98, 0.000171299060741559732192347 34,
0.000138476237460891912810570 51, 0.000138351419709315632492613 31, 0.000171406401870234469239293 04,
0.000103162807998310334820987 74, 0.000178715869246580746779531 90, 0.000154008944911718362097072 24,
0.000093005161075859527249065 64, 0.000124671838859857525995706 55, 0.000124372087781061073892428 7833,
0.000154293068207665218585789 01, 0.000092731124393513004251950 18, 0.000178938910526219621111345 011,
0.000061061746788211175552239 80, 0.000195402576269580676056325 52, 0.000030188442536011951441684 08,
0.000201166220366356067394608 201, 0.000030333995476459733609811 07, 0.000195279733697034386637539 36,
0.000061279814047253834251999 25, 0.000051844626284183320801424 86, 0.000025629350044781963282496 46,
0.000166106424017364824119794 01, 0.000170960963149568078919826 60, 0.000025820484000143181171952 945,
0.000165919824881411814669618 38, 0.000052151522936473897684109 96, 0.000151816502240506844565240 41,
0.000079137880286768932133582 84, 0.000130806666348657181725770 32, 0.000106062017184663579102629 76,

0.00010561992236396534970265665, 0.00013123095997358502845867023, 0.00007874056746028607154264806,
0.00015215293565708012404023852, 0.00006222040683111706386304585, 0.00008335959549927594771506767,
0.00010258632797025758385221772, 0.00008281781030143020561583300, 0.00010310196836561219915252841,
0.00006173032279876991220736449, 0.00011949298694665030104531539, 0.00004063789668697304286251737,
0.00013040326174395598201986857, 0.00002008545666738634850921955, 0.00013417078693930139946313695,
0.00002031544022311951151228693, 0.00013017928887589910106996338, 0.00004101619549416230968083563,
0.00011908714891399693085790451, 0.00008966129511665799339120708, 0.00009220505754501783900883849,
0.00001403056240262716242129474, 0.00001376989274697555899675417, 0.00008942866614871386606554216,
0.00002830234007726331423157248, 0.00008178486589571475084449459, 0.00004290405945425181214207197,
0.00007043310183359100635439726, 0.00005743847000855366427595275, 0.00005684299892959915643598015,
0.00007098765321529926426193102, 0.00004235476262650960586854040, 0.00008221259424061854988261683,
0.00002787253843428859665471917, 0.00003558242376847433067945520, 0.00002869342762609467504695757,
0.00002930084053037463522202975, 0.00003613072621715948874081617, 0.00002135823672786075472077895,
0.00004175360325406188895941417, 0.00001403897976335879906484988, 0.00004545334004177860255831167,
0.00000692532427039974761851914, 0.00004668015966986112082279942, 0.00000720767128861170651557338,
0.00004523500843280730523889716, 0.00001449990051279543850531236, 0.00004134314658568343837669740,
0.00002193448472471466867070311, -0.00000018536275476188765807097, -0.00000063555088312147990051593,
-0.00000069105112124982154233108, 0.00000010937065147812057006363, 0.00000014112585398400017200403,
-0.00000067819484557037945251883, 0.00000012223552315333760713112, -0.00000062520755138185074585962,
0.00000003733967313262340997177, -0.00000054985479003812424010330, -0.00000011406121089387746799567,
-0.00000045335505234210048940157, -0.00000030872975031862248642604, -0.00000033173271494498099654726,
-0.00000049961619875163898049847, -0.00002976391214334230916720159, -0.00002224332488489791864001363,
-0.00003633824795277500798460282, -0.00004233538267895286426930473, -0.00001468822620388765633799976,
-0.00004640416296459308508667169, -0.00000728913626897230496049212, -0.00004789692346398960015292981,
-0.00000699111979228816111486648, -0.00004656460671813850821752767, -0.00001422027357655865031330240,
-0.00004265386185347319115447606, -0.00002168971741222897682468268, -0.00003679552505313648966224147,
-0.00002921817133760940435696120, -0.00001382053872145808158609231, -0.00002803080423196321196988023,
-0.00009060990761493040927636311, -0.00008296424221490546820501760, -0.00004265961777418032834169673,
-0.00007153758346569133279302388, -0.00005733533440305722567609942, -0.00005782780795828509516143603,
-0.00007114058714583946195497699, -0.00004317621071903213610066949, -0.00008269851606185571078408425,
-0.00002847984121712703668954751, -0.00009048023917856811966595881, -0.00001411327712725095190880210,
-0.00009327121660228933602409390, -0.00010318331096309616562445299, -0.00011984926043979200348168351,
-0.00006244976420704460807684089, -0.00004117762498992887174997096, -0.00013103893622691932877022403,
-0.00002039581194293668963161623, -0.00013502106123139932542659425, -0.00002011614821639516964232354,
-0.00013114060303592856016052792, -0.00004075872687014014603135159, -0.00012006568355820456989353773,
-0.00006197987877876467933870724, -0.00010352074553398723177032037, -0.00008323192631180004077799672,
-0.00008366647472881513195215597, -0.00016661982395304531716188901, -0.00015254720284414819103484262,
-0.00005191459148860947298053173, -0.00007890803977945857377099270, -0.00013152454305630307466838669,
-0.00010591450193005250970598730, -0.00010629131245839624965877218, -0.00013124084510660762969806725,
-0.00007932537509100781797973662, -0.00015237219495052033743469344, -0.00005229449907647272581491180,
-0.00016654075739959696661300625, -0.00002589553811933865517776535, -0.00017156478127045330598493544,
-0.00002563586837672157138761503, -0.00009315588308157499923371769, -0.00006140378828223462081940809,
-0.00017902977521352931135692477, -0.00019563886207072810237417304, -0.00003040125774207811000515722,
-0.00020151795372538009896642985, -0.00003016758948722044825696206, -0.00019570217940511938615154308,
-0.00006106996966507975167781641, -0.00017917336493616733169957633, -0.00009279481635922343226927089,
-0.00015447997982178347862840251, -0.00012451474238563082835504203, -0.00012483612445501407950051997,
-0.00015424205974566204940118253, -0.00006787379414957882630982794, -0.00010310846865362845376553474,
-0.00019892317488940359734329866, -0.00017150610400596397794176218, -0.00013832132833937183760859846,
-0.00013858878543495013356198764, -0.00017130791711928515627834380, -0.00010341001070076125513755305,
-0.00019880316650244890031885847, -0.00006815557561437242427900690, -0.00021722025818499458046755002,
-0.00003373987825876595495486546, -0.00022373374742836631742098907, -0.00003353732141415467825438373,
-0.00021727331389793976402884101, -0.00021093144903028035572374599, -0.00023045009102277296854366373,
-0.00007229013674218415913801800, -0.00003578305871748360640150280, -0.00023734955778653999964970811,
-0.00003561583216870238619965877, -0.00023049492413110690388136637, -0.00007206567010067558643263141,
-0.00021103047189652274348223293, -0.00010945482579816623021438243, -0.00018194432559134879529884332,
-0.00014680650550965147586868209, -0.00014701869566590132997439822, -0.00018178460549161468487455728,
-0.00010969288485332497659185264, -0.00021502053205313794098998970, -0.00018538753811050880798694251,
-0.00011159209960883873999359928, -0.00014964640440136669271437020, -0.00014979879388047950741075232,

-0.0001852698713089982632089280 9, -0.0001117616283369281585634291 7, -0.0002149448733316481631970551 5,
-0.0000736480306384209286061845 5, -0.0002348114949726790804535037 1, -0.0000364521462086874505008934,
-0.0002418304330965812291506789 0, -0.0000363235893945087213031779 1, -0.0002348469732821895777918896 9,
-0.0000734857007016584716030741 0, -0.0000721763745909064516554157 2, -0.0000357212313208313717136262 2,
-0.0002301319389389389347772317123 7, -0.0002369975032134581747339519 0, -0.0000356337504856635867178139 8,
-0.0002301554146367136436182598 8, -0.0000720805141500003535449253 6, -0.0002107344128860830624692251 9,
-0.0001094401502896924028088165 8, -0.0001816994787428428232648608 4, -0.0001467338842867341619936177 9,
-0.0001468197436736434732994044270, -0.0001816303959144190951699449 1, -0.0001095353015829042124832867 8,
-0.0002106871475831943740526092 1, -0.0001710214372401646352496451 5, -0.0001381948502711113707045603 7,
-0.0001381832439139102191597209748, -0.0001710086294984781648837596 2, -0.0001030982484423041722117891 9,
-0.0001983245950921147396708854 6, -0.0000679309619564844616125196 0, -0.0002165939481522992458156773 3,
-0.0000336181721734709240798251 5, -0.0002230388884276856957013246 4, -0.0000335731670255716083128838 5,
-0.0002166023846998158235849230 0, -0.0000679052145153201744268503 5, -0.0001983375749476203107717065 0,
-0.0001030835509166814307688286 9, -0.0000302212927613724981115023 2, -0.0000302235419158374793479129 9,
-0.0002005022423614099928712573 0, -0.0001947179923671681620912787 8, -0.0000611211945928302795511336 0,
-0.0001783121022944914231024571 3, -0.0000927684317912612105717223 8, -0.0001537658063308078304487480 6,
-0.0001243261871340634689905857 4, -0.0001242553096574600286772227 5, -0.0001538179298287123387099784 0,
-0.0000926968620441243106380935 2, -0.0001783402973946208417040104 6, -0.0000610741955648915157929199 1,
-0.0001947283373796509869067816 7, -0.0001056987251690965956595416 6, -0.0001307254272156473454136183 0,
-0.0001055363799722574213134362 9, -0.0000787298176988141749357494 3, -0.0001515125666933500646425120 3,
-0.0000518688619457179580390406 0, -0.0001653886820549496379072901 3, -0.0000256675722134768443999288 5,
-0.0001702671579400425426464860 1, -0.0000257071150885116235680887 7, -0.0001653536894502248306027675 2,
-0.0000519898339882711359866140 9, -0.0001514333485586396861165975 3, -0.0000788928378499531161903782 2,
-0.0001305975034620948053293298 9, -0.0001335034006622985442835210 8, -0.0001296427104874910141152355 4,
-0.0000202041576990349337652852 7, -0.0000408623660794125683273780 9, -0.0001187356937352174998450901 6,
-0.0000619911582370924252099672 6, -0.0001024061257397207815641321 2, -0.0000830196017473515586199833 7,
-0.0000827559570046525427812131 3, -0.0001026240543308132844614599 7, -0.0000617326684932345206993989 4,
-0.0001188804090637931063596452 5, -0.0000406680820770553345358706 8, -0.0001297116116179091967804820 6,
-0.0000201251750552886187113112 2, -0.0000567870960491941213386588 0, -0.0000423573648926948504367037 7,
-0.0000705961110059211160473735 5, -0.0000817008545544848399340767 7, -0.0000279018337021662385811354 7,
-0.0000890703754115873005091363 9, -0.0000138092416983661885672863 4, -0.0000916239455557558656728990 2,
-0.0000139238087683264939682431 2, -0.0000889552517800508467729625 5, -0.0000281662615922258787154019 3,
-0.0000814721791183001974772162 6, -0.0000427130918372246290480934 2, -0.0000702716235077951908430046 7,
-0.0000571615056859905220233014 6, -0.0000071072018685230503701232 4, -0.0000143887387563767801216142 0,
-0.0000448488068001396376644623 9, -0.0000410690390510249218642031 8, -0.0000217983492173407751915908 3,
-0.0000354246934009980221050480 1, -0.0000291166432970444047205211 4, -0.0000286254897526302296516514 3,
-0.0000358704645744766198282860 2, -0.0000213476550918627314460361 6, -0.0000413999313221732789433958 4,
-0.0000140605458705331124238946 4, -0.0000450228544515788036035897 1, -0.0000069624918963301015245007 3,
-0.0000462329109318980420234232 5, 0.000000224679115270076657042 46, 0.000000481674926062977296473 19,
0.000000488995186273133720064 63, 0.000000323634629591537532302 31, 0.000000747066687273364509514 79,
0.000000151861058105160951307 98, 0.000000933699330495662247969 36, -0.000000015975259314781658701 29,
0.000000988500669697860421977 72, -0.000000005809251968019942385 924, 0.000000925770029497265418898 61,
-0.000000048200787602423613798 04, 0.000000797976233256873854820 19, 0.000000041452432646459117052 69,
0.000000647374912147137865058 30, 0.000046800357440172450629572 30, 0.000042899674371040838569569 40,
0.000014276460836239867287127 13, 0.000021704914700296568981366 73, 0.000037003336859304344268282 31,
0.000029198343676355583656712 58, 0.000029904953308716396439460 43, 0.000036311309087926970393198 77,
0.000022312166479834413498116 37, 0.000042345859536182621485605 33, 0.000014697270056984819468432 27,
0.000046492102782998422387181 20, 0.000007260225996523572481974 09, 0.000048072559897832391987305 27,
0.000007077741937156103082932 38, 0.000043281010739282734935227 33, 0.000028506991064106836862015 67,
0.000082595090568763512330063 84, 0.000090465043098142102276360 00, 0.000014089177337854421600822 46,
0.000093379343382774256479041 12, 0.000013901727857900301966922 32, 0.000090828056486411189000829 14,
0.000028065639601360810025196 95, 0.000083237027969463729779216 19, 0.000042628728537560810652341 10,
0.000071795791719898895675601 45, 0.000057240085821183764986340 44, 0.000058019295258296322453147 46,
0.000071012090438072318046743 01, 0.000120383338342868142947610 09, 0.000103833067640684989849725 83,
0.000061922406698916874828383 94, 0.000083094695256658750778105 96, 0.000083905787926077225009555 14,
0.000103001218127580395538325 79, 0.000062587158758185183590069 60145, 0.000119691063903905567139714 09,
0.000041220965335329754018062 20, 0.000130983031259100603151818 15, 0.000020375998585039873734273 13,
0.000135116353057822600027340 63, 0.000020194070807821726991695 97, 0.000131378573240537206701095 13,

```
0.00004078093529010286559435133, 0.00005234954471178215960457808, 0.00002587883134899454044136613,
0.00016649349418400413627011636, 0.00017168185862034981243803844, 0.00002571317762172624241329491,
0.00016689317445149211391504374, 0.00005193466288569443666412601, 0.00015290857304592109277077694,
0.00007884675684891602855716014, 0.00013188042058524072496113189, 0.00010577246912910297216552785,
0.00010656630312335742445401660, 0.00013105468082139063346232277, 0.00007948646572211354763108637,
0.00015221385474413457556883134, 0.00009275223431934548432714616, 0.00012440355572374795184269203,
0.00015485901300309637775888538, 0.00012512830010899064921668644, 0.00015409710537769833295709820,
0.00009332802808153728359348567, 0.00017891747244875335883310119, 0.00006146429238704890509435719,
0.00019563624679433259164612613, 0.00003038600629161280648467096, 0.00020167483423115019087329403,
0.00003024702323752094363881228, 0.00019600983424611289154294858, 0.00006109853669347578390691478,
0.00017956355425588266033418150, 0.00021729386764661110095428476, 0.00022394623173360359274369769,
0.00003362128005204847520015141, 0.00003372421628176038894833880, 0.00021761287056678890782304703,
0.00006792007336981402225700027, 0.00019932597152122770358534387, 0.00010310376552645847273229074,
0.00017188644896625095018563578, 0.00013827097185156560221752642, 0.00013887814927907385104816074,
0.00017124231788059795437810517, 0.00010357934222891635389022374, 0.00019877640983226803924992232,
0.00006821455257841429445538062, 0.00018230925383158519165367750, 0.00014728808845052273408977872,
0.00014683821948869725635029859, 0.00018182656923916606147434427, 0.00010984671095598225567298994,
0.00021102242965774522750511955, 0.00007234097598022849436374959, 0.00023063008265034889185222944,
0.00003576519103914519034058406, 0.00023763783581554187422126001, 0.00003570603085499142020356120,
0.00023087139189595877854579986, 0.00007213616985492927720455963, 0.00021143688368655628919551281,
0.00010950145260021515328328207]]
```

Definition at line 60 of file testCase.py.

6.10.2.3 calcWeightsGaussCorrect

```
list testCase.calcWeightsGaussCorrect = [[-0.57735026918962573105886 80, 0.57735026918962573105886 80],[1.↩
00000000000000044408920 99, 1.00000000000000044408920 99]]
```

Definition at line 52 of file testCase.py.

6.10.2.4 conducterThickness

```
float testCase.conducterThickness = 0.005
```

Definition at line 15 of file testCase.py.

6.10.2.5 gaussOrder

```
int testCase.gaussOrder = 2
```

Definition at line 12 of file testCase.py.

6.10.2.6 gaußLegendreCorrect

```
list testCase.gaußLegendreCorrect = [[0.21132486540518713, 0.21132486540518713, 0.7886751345948129,
0.7886751345948129], [0.16666666666666669, 0.6220084679281462, 0.044658198738520456, 0.↩
16666666666666669], [0.19716878364870338, 0.19716878364870338, 0.052831216351296825, 0.↩
052831216351296825]]
```

Definition at line 54 of file testCase.py.

6.10.2.7 matElementsShouldGetValueCorrect

```
list testCase.matElementsShouldGetValueCorrect = [0.0000000000, 1.0000000000, 2.0000000000,
3.0000000000, 4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000, 9.0000000000,
10.0000000000, 11.0000000000, 12.0000000000, 13.0000000000, 14.0000000000, 15.0000000000,
16.0000000000, 17.0000000000, 18.0000000000, 19.0000000000, 20.0000000000, 21.0000000000,
22.0000000000, 23.0000000000, 24.0000000000, 25.0000000000, 26.0000000000, 27.0000000000,
28.0000000000, 29.0000000000, 30.0000000000, 31.0000000000, 32.0000000000, 33.0000000000,
34.0000000000, 35.0000000000, 36.0000000000, 37.0000000000, 38.0000000000, 39.0000000000,
40.0000000000, 41.0000000000, 42.0000000000, 43.0000000000, 44.0000000000, 45.0000000000,
46.0000000000, 47.0000000000, 48.0000000000, 49.0000000000, 50.0000000000, 51.0000000000,
52.0000000000, 53.0000000000, 54.0000000000, 55.0000000000, 56.0000000000, 57.0000000000,
58.0000000000, 59.0000000000, 60.0000000000, 61.0000000000, 62.0000000000, 63.0000000000,
64.0000000000, 65.0000000000, 66.0000000000, 67.0000000000, 68.0000000000, 69.0000000000,
70.0000000000, 71.0000000000, 72.0000000000, 73.0000000000, 74.0000000000, 75.0000000000,
76.0000000000, 77.0000000000, 78.0000000000, 79.0000000000, 80.0000000000, 81.0000000000,
82.0000000000, 83.0000000000, 84.0000000000, 85.0000000000, 86.0000000000, 87.0000000000,
88.0000000000, 89.0000000000, 90.0000000000, 91.0000000000, 92.0000000000, 93.0000000000,
94.0000000000, 95.0000000000, 96.0000000000, 97.0000000000, 98.0000000000, 99.0000000000,
100.0000000000, 101.0000000000, 102.0000000000, 103.0000000000, 104.0000000000, 105.0000000000,
106.0000000000, 107.0000000000, 108.0000000000, 109.0000000000, 110.0000000000, 111.0000000000,
112.0000000000, 113.0000000000, 114.0000000000, 115.0000000000, 116.0000000000, 117.0000000000,
118.0000000000, 119.0000000000, 120.0000000000, 121.0000000000, 122.0000000000, 123.0000000000,
124.0000000000, 125.0000000000, 126.0000000000, 127.0000000000, 128.0000000000, 129.0000000000,
130.0000000000, 131.0000000000, 132.0000000000, 133.0000000000, 134.0000000000, 135.0000000000,
136.0000000000, 137.0000000000, 138.0000000000, 139.0000000000, 140.0000000000, 141.0000000000,
142.0000000000, 143.0000000000, 144.0000000000, 145.0000000000, 146.0000000000, 147.0000000000,
148.0000000000, 149.0000000000, 150.0000000000, 151.0000000000, 152.0000000000, 153.0000000000,
154.0000000000, 155.0000000000, 156.0000000000, 157.0000000000, 158.0000000000, 159.0000000000,
160.0000000000, 161.0000000000, 162.0000000000, 163.0000000000, 164.0000000000, 165.0000000000,
166.0000000000, 167.0000000000, 168.0000000000, 169.0000000000, 170.0000000000, 171.0000000000,
172.0000000000, 173.0000000000, 174.0000000000, 175.0000000000, 176.0000000000, 177.0000000000,
178.0000000000, 179.0000000000, 180.0000000000, 181.0000000000, 182.0000000000, 183.0000000000,
184.0000000000, 185.0000000000, 186.0000000000, 187.0000000000, 188.0000000000, 189.0000000000,
190.0000000000, 191.0000000000, 192.0000000000, 193.0000000000, 194.0000000000, 195.0000000000,
196.0000000000, 197.0000000000, 198.0000000000, 199.0000000000, 200.0000000000, 201.0000000000,
202.0000000000, 203.0000000000, 204.0000000000, 205.0000000000, 206.0000000000, 207.0000000000,
208.0000000000, 209.0000000000, 210.0000000000, 211.0000000000, 212.0000000000, 213.0000000000,
214.0000000000, 215.0000000000, 216.0000000000, 217.0000000000, 218.0000000000, 219.0000000000,
220.0000000000, 221.0000000000, 222.0000000000, 223.0000000000, 224.0000000000, 225.0000000000,
226.0000000000, 227.0000000000, 228.0000000000, 229.0000000000, 230.0000000000, 231.0000000000,
232.0000000000, 233.0000000000, 234.0000000000, 235.0000000000, 236.0000000000, 237.0000000000,
238.0000000000, 239.0000000000, 240.0000000000, 241.0000000000, 242.0000000000, 243.0000000000,
244.0000000000, 245.0000000000, 246.0000000000, 247.0000000000, 248.0000000000, 249.0000000000,
250.0000000000, 251.0000000000, 252.0000000000, 253.0000000000, 254.0000000000, 255.0000000000,
```

256.0000000000, 257.0000000000, 258.0000000000, 259.0000000000, 260.0000000000, 261.0000000000,
262.0000000000, 263.0000000000, 1.0000000000, 2.0000000000, 0.0000000000, 6.0000000000, 8.↩
0000000000, 9.0000000000, 3.0000000000, 2.0000000000, 7.0000000000, 3.0000000000, 1.0000000000,
5.0000000000, 4.0000000000, 1.0000000000, 9.0000000000, 4.0000000000, 2.0000000000, 12.↩
0000000000, 10.0000000000, 18.0000000000, 12.0000000000, 2.0000000000, 5.0000000000, 3.↩
0000000000, 10.0000000000, 4.0000000000, 11.0000000000, 14.0000000000, 8.0000000000, 15.↩
0000000000, 7.0000000000, 1.0000000000, 14.0000000000, 6.0000000000, 0.0000000000, 1.0000000000,
6.0000000000, 15.0000000000, 16.0000000000, 17.0000000000, 1.0000000000, 23.0000000000, 9.↩
0000000000, 12.0000000000, 17.0000000000, 1.0000000000, 3.0000000000, 8.0000000000, 18.↩
0000000000, 78.0000000000, 4.0000000000, 5.0000000000, 81.0000000000, 77.0000000000, 11.↩
0000000000, 77.0000000000, 10.0000000000, 18.0000000000, 9.0000000000, 17.0000000000, 4.↩
0000000000, 3.0000000000, 19.0000000000, 21.0000000000, 7.0000000000, 14.0000000000, 15.↩
0000000000, 22.0000000000, 21.0000000000, 13.0000000000, 7.0000000000, 6.0000000000, 14.↩
0000000000, 29.0000000000, 8.0000000000, 6.0000000000, 22.0000000000, 23.0000000000, 8.↩
0000000000, 30.0000000000, 17.0000000000, 23.0000000000, 25.0000000000, 24.0000000000, 9.↩
0000000000, 12.0000000000, 25.0000000000, 16.0000000000, 27.0000000000, 8.0000000000, 19.↩
0000000000, 26.0000000000, 12.0000000000, 10.0000000000, 4.0000000000, 81.0000000000, 85.↩
0000000000, 19.0000000000, 26.0000000000, 33.0000000000, 27.0000000000, 12.0000000000, 18.↩
0000000000, 17.0000000000, 21.0000000000, 13.0000000000, 14.0000000000, 20.0000000000, 28.↩
0000000000, 13.0000000000, 144.0000000000, 143.0000000000, 22.0000000000, 15.0000000000, 28.↩
0000000000, 14.0000000000, 35.0000000000, 29.0000000000, 21.0000000000, 29.0000000000, 30.↩
0000000000, 8.0000000000, 15.0000000000, 16.0000000000, 37.0000000000, 30.0000000000, 31.↩
0000000000, 25.0000000000, 16.0000000000, 32.0000000000, 16.0000000000, 17.0000000000, 31.↩
0000000000, 27.0000000000, 24.0000000000, 85.0000000000, 18.0000000000, 19.0000000000, 33.↩
0000000000, 90.0000000000, 33.0000000000, 32.0000000000, 17.0000000000, 25.0000000000, 19.↩
0000000000, 34.0000000000, 151.0000000000, 22.0000000000, 35.0000000000, 147.0000000000, 144.↩
0000000000, 21.0000000000, 30.0000000000, 35.0000000000, 36.0000000000, 22.0000000000, 23.↩
0000000000, 15.0000000000, 29.0000000000, 43.0000000000, 37.0000000000, 23.0000000000, 24.↩
0000000000, 36.0000000000, 16.0000000000, 37.0000000000, 24.0000000000, 32.0000000000, 39.↩
0000000000, 38.0000000000, 25.0000000000, 39.0000000000, 41.0000000000, 31.0000000000, 25.↩
0000000000, 34.0000000000, 27.0000000000, 19.0000000000, 27.0000000000, 34.0000000000, 90.↩
0000000000, 40.0000000000, 26.0000000000, 41.0000000000, 40.0000000000, 47.0000000000, 32.↩
0000000000, 27.0000000000, 33.0000000000, 36.0000000000, 29.0000000000, 151.0000000000, 22.↩
0000000000, 42.0000000000, 28.0000000000, 42.0000000000, 29.0000000000, 35.0000000000, 30.↩
0000000000, 43.0000000000, 43.0000000000, 44.0000000000, 24.0000000000, 30.0000000000, 31.↩
0000000000, 38.0000000000, 31.0000000000, 39.0000000000, 37.0000000000, 46.0000000000, 45.↩
0000000000, 44.0000000000, 46.0000000000, 48.0000000000, 41.0000000000, 38.0000000000, 31.↩
0000000000, 32.0000000000, 34.0000000000, 90.0000000000, 33.0000000000, 96.0000000000, 47.↩
0000000000, 103.0000000000, 34.0000000000, 54.0000000000, 47.0000000000, 48.0000000000, 32.↩
0000000000, 39.0000000000, 35.0000000000, 43.0000000000, 156.0000000000, 36.0000000000, 162.↩
0000000000, 49.0000000000, 151.0000000000, 50.0000000000, 36.0000000000, 37.0000000000, 56.↩
0000000000, 44.0000000000, 30.0000000000, 42.0000000000, 49.0000000000, 51.0000000000, 38.↩
0000000000, 43.0000000000, 57.0000000000, 45.0000000000, 37.0000000000, 50.0000000000, 52.↩
0000000000, 51.0000000000, 46.0000000000, 44.0000000000, 38.0000000000, 39.0000000000, 38.↩
0000000000, 52.0000000000, 53.0000000000, 45.0000000000, 48.0000000000, 34.0000000000, 103.↩
0000000000, 40.0000000000, 54.0000000000, 41.0000000000, 39.0000000000, 53.0000000000, 55.↩
0000000000, 41.0000000000, 54.0000000000, 46.0000000000, 61.0000000000, 42.0000000000, 162.↩
0000000000, 43.0000000000, 56.0000000000, 57.0000000000, 63.0000000000, 56.0000000000, 44.↩
0000000000, 43.0000000000, 45.0000000000, 44.0000000000, 52.0000000000, 58.0000000000, 57.↩
0000000000, 65.0000000000, 45.0000000000, 46.0000000000, 51.0000000000, 60.0000000000, 59.↩
0000000000, 58.0000000000, 53.0000000000, 46.0000000000, 55.0000000000, 52.0000000000, 60.↩
0000000000, 48.0000000000, 48.0000000000, 109.0000000000, 61.0000000000, 47.0000000000, 103.↩
0000000000, 115.0000000000, 41.0000000000, 60.0000000000, 62.0000000000, 61.0000000000, 48.↩
0000000000, 53.0000000000, 50.0000000000, 49.0000000000, 43.0000000000, 63.0000000000, 169.↩
0000000000, 162.0000000000, 58.0000000000, 51.0000000000, 63.0000000000, 64.0000000000, 50.↩
0000000000, 44.0000000000, 51.0000000000, 57.0000000000, 65.0000000000, 52.0000000000, 64.↩
0000000000, 67.0000000000, 72.0000000000, 52.0000000000, 66.0000000000, 65.0000000000, 60.↩
0000000000, 52.0000000000, 53.0000000000, 59.0000000000, 67.0000000000, 62.0000000000, 55.↩

0000000000, 121.0000000000, 68.0000000000, 62.0000000000, 115.0000000000, 48.0000000000, 55.↩
0000000000, 54.0000000000, 55.0000000000, 68.0000000000, 61.0000000000, 60.0000000000, 69.↩
0000000000, 67.0000000000, 64.0000000000, 56.0000000000, 57.0000000000, 50.0000000000, 175.↩
0000000000, 70.0000000000, 169.0000000000, 58.0000000000, 57.0000000000, 71.0000000000, 70.↩
0000000000, 63.0000000000, 65.0000000000, 52.0000000000, 64.0000000000, 58.0000000000, 59.↩
0000000000, 72.0000000000, 75.0000000000, 71.0000000000, 59.0000000000, 72.0000000000, 66.↩
0000000000, 62.0000000000, 59.0000000000, 60.0000000000, 127.0000000000, 69.0000000000, 73.↩
0000000000, 121.0000000000, 61.0000000000, 62.0000000000, 68.0000000000, 67.0000000000, 62.↩
0000000000, 63.0000000000, 71.0000000000, 64.0000000000, 181.0000000000, 187.0000000000, 175.↩
0000000000, 74.0000000000, 74.0000000000, 64.0000000000, 70.0000000000, 65.0000000000, 75.↩
0000000000, 75.0000000000, 65.0000000000, 59.0000000000, 68.0000000000, 69.0000000000, 76.↩
0000000000, 75.0000000000, 71.0000000000, 187.0000000000, 193.0000000000, 70.0000000000, 74.↩
0000000000, 65.0000000000, 76.0000000000, 71.0000000000, 193.0000000000, 74.0000000000, 10.↩
0000000000, 78.0000000000, 79.0000000000, 10.0000000000, 84.0000000000, 77.0000000000, 81.↩
0000000000, 80.0000000000, 79.0000000000, 83.0000000000, 80.0000000000, 82.0000000000, 78.↩
0000000000, 79.0000000000, 82.0000000000, 84.0000000000, 78.0000000000, 18.0000000000, 10.↩
0000000000, 84.0000000000, 85.0000000000, 78.0000000000, 88.0000000000, 86.0000000000, 80.↩
0000000000, 83.0000000000, 87.0000000000, 84.0000000000, 79.0000000000, 86.0000000000, 82.↩
0000000000, 82.0000000000, 81.0000000000, 89.0000000000, 88.0000000000, 80.0000000000, 85.↩
0000000000, 78.0000000000, 81.0000000000, 18.0000000000, 84.0000000000, 26.0000000000, 90.↩
0000000000, 89.0000000000, 91.0000000000, 87.0000000000, 92.0000000000, 82.0000000000, 93.↩
0000000000, 82.0000000000, 91.0000000000, 88.0000000000, 86.0000000000, 89.0000000000, 93.↩
0000000000, 87.0000000000, 82.0000000000, 94.0000000000, 84.0000000000, 90.0000000000, 84.↩
0000000000, 88.0000000000, 95.0000000000, 85.0000000000, 94.0000000000, 89.0000000000, 40.↩
0000000000, 102.0000000000, 26.0000000000, 33.0000000000, 95.0000000000, 85.0000000000, 96.↩
0000000000, 93.0000000000, 92.0000000000, 97.0000000000, 99.0000000000, 86.0000000000, 87.↩
0000000000, 98.0000000000, 91.0000000000, 97.0000000000, 87.0000000000, 99.0000000000, 100.↩
0000000000, 91.0000000000, 94.0000000000, 88.0000000000, 105.0000000000, 89.0000000000, 88.↩
0000000000, 106.0000000000, 100.0000000000, 101.0000000000, 93.0000000000, 95.0000000000,
101.0000000000, 102.0000000000, 90.0000000000, 89.0000000000, 94.0000000000, 103.0000000000,
108.0000000000, 40.0000000000, 90.0000000000, 102.0000000000, 209.0000000000, 210.0000000000,
104.0000000000, 92.0000000000, 213.0000000000, 99.0000000000, 98.0000000000, 91.0000000000,
97.0000000000, 209.0000000000, 104.0000000000, 105.0000000000, 93.0000000000, 97.0000000000,
91.0000000000, 94.0000000000, 93.0000000000, 105.0000000000, 106.0000000000, 102.0000000000,
106.0000000000, 94.0000000000, 95.0000000000, 112.0000000000, 107.0000000000, 101.0000000000,
107.0000000000, 96.0000000000, 113.0000000000, 95.0000000000, 90.0000000000, 108.0000000000,
108.0000000000, 109.0000000000, 114.0000000000, 47.0000000000, 96.0000000000, 40.0000000000,
54.0000000000, 110.0000000000, 99.0000000000, 213.0000000000, 97.0000000000, 219.0000000000,
105.0000000000, 100.0000000000, 106.0000000000, 99.0000000000, 104.0000000000, 93.0000000000,
110.0000000000, 116.0000000000, 111.0000000000, 105.0000000000, 100.0000000000, 112.0000000000,
111.0000000000, 117.0000000000, 94.0000000000, 101.0000000000, 113.0000000000, 118.0000000000,
102.0000000000, 101.0000000000, 112.0000000000, 103.0000000000, 113.0000000000, 114.0000000000,
102.0000000000, 96.0000000000, 114.0000000000, 115.0000000000, 103.0000000000, 54.0000000000,
120.0000000000, 116.0000000000, 219.0000000000, 105.0000000000, 225.0000000000, 104.0000000000,
105.0000000000, 116.0000000000, 117.0000000000, 106.0000000000, 106.0000000000, 107.0000000000,
101.0000000000, 118.0000000000, 117.0000000000, 123.0000000000, 118.0000000000, 108.0000000000,
107.0000000000, 114.0000000000, 102.0000000000, 119.0000000000, 113.0000000000, 120.0000000000,
108.0000000000, 109.0000000000, 119.0000000000, 103.0000000000, 120.0000000000, 121.0000000000,
61.0000000000, 54.0000000000, 109.0000000000, 117.0000000000, 122.0000000000, 110.0000000000,
225.0000000000, 230.0000000000, 111.0000000000, 105.0000000000, 116.0000000000, 123.0000000000,
122.0000000000, 111.0000000000, 112.0000000000, 106.0000000000, 123.0000000000, 107.0000000000,
112.0000000000, 119.0000000000, 124.0000000000, 113.0000000000, 114.0000000000, 130.0000000000,
125.0000000000, 124.0000000000, 113.0000000000, 120.0000000000, 118.0000000000, 131.0000000000,
115.0000000000, 114.0000000000, 126.0000000000, 125.0000000000, 121.0000000000, 109.0000000000,
119.0000000000, 61.0000000000, 132.0000000000, 115.0000000000, 126.0000000000, 120.0000000000,
127.0000000000, 68.0000000000, 123.0000000000, 116.0000000000, 230.0000000000, 235.0000000000,
117.0000000000, 128.0000000000, 122.0000000000, 117.0000000000, 112.0000000000, 124.0000000000,
118.0000000000, 133.0000000000, 128.0000000000, 129.0000000000, 129.0000000000, 118.0000000000,

123.0000000000, 130.0000000000, 119.0000000000, 130.0000000000, 120.0000000000, 119.0000000000,
131.0000000000, 131.0000000000, 121.0000000000, 120.0000000000, 132.0000000000, 73.0000000000,
68.0000000000, 121.0000000000, 133.0000000000, 235.0000000000, 240.0000000000, 122.0000000000,
123.0000000000, 130.0000000000, 123.0000000000, 133.0000000000, 137.0000000000, 124.0000000000,
134.0000000000, 124.0000000000, 134.0000000000, 135.0000000000, 125.0000000000, 119.0000000000,
129.0000000000, 131.0000000000, 136.0000000000, 132.0000000000, 139.0000000000, 135.0000000000,
130.0000000000, 125.0000000000, 120.0000000000, 126.0000000000, 126.0000000000, 121.0000000000,
131.0000000000, 127.0000000000, 240.0000000000, 128.0000000000, 137.0000000000, 123.0000000000,
129.0000000000, 137.0000000000, 129.0000000000, 130.0000000000, 138.0000000000, 135.0000000000,
130.0000000000, 141.0000000000, 138.0000000000, 134.0000000000, 139.0000000000, 131.0000000000,
131.0000000000, 132.0000000000, 138.0000000000, 140.0000000000, 240.0000000000, 129.0000000000,
133.0000000000, 134.0000000000, 250.0000000000, 245.0000000000, 137.0000000000, 141.0000000000,
135.0000000000, 134.0000000000, 140.0000000000, 136.0000000000, 131.0000000000, 135.0000000000,
255.0000000000, 138.0000000000, 141.0000000000, 142.0000000000, 137.0000000000, 250.0000000000,
139.0000000000, 140.0000000000, 135.0000000000, 138.0000000000, 140.0000000000, 141.0000000000,
21.0000000000, 144.0000000000, 20.0000000000, 146.0000000000, 143.0000000000, 28.0000000000,
21.0000000000, 146.0000000000, 147.0000000000, 146.0000000000, 149.0000000000, 143.0000000000,
149.0000000000, 145.0000000000, 144.0000000000, 150.0000000000, 147.0000000000, 143.0000000000,
150.0000000000, 28.0000000000, 151.0000000000, 146.0000000000, 144.0000000000, 149.0000000000,
145.0000000000, 145.0000000000, 152.0000000000, 148.0000000000, 153.0000000000, 154.0000000000,
146.0000000000, 150.0000000000, 154.0000000000, 155.0000000000, 147.0000000000, 149.0000000000,
151.0000000000, 160.0000000000, 146.0000000000, 156.0000000000, 42.0000000000, 35.0000000000,
155.0000000000, 28.0000000000, 147.0000000000, 150.0000000000, 148.0000000000, 149.0000000000,
153.0000000000, 157.0000000000, 158.0000000000, 159.0000000000, 154.0000000000, 152.0000000000,
149.0000000000, 149.0000000000, 159.0000000000, 153.0000000000, 150.0000000000, 166.0000000000,
160.0000000000, 160.0000000000, 161.0000000000, 156.0000000000, 151.0000000000, 150.0000000000,
155.0000000000, 162.0000000000, 161.0000000000, 151.0000000000, 168.0000000000, 42.0000000000,
164.0000000000, 153.0000000000, 158.0000000000, 152.0000000000, 157.0000000000, 159.0000000000,
165.0000000000, 153.0000000000, 164.0000000000, 154.0000000000, 158.0000000000, 165.0000000000,
153.0000000000, 171.0000000000, 166.0000000000, 167.0000000000, 154.0000000000, 155.0000000000,
150.0000000000, 166.0000000000, 161.0000000000, 155.0000000000, 168.0000000000, 167.0000000000,
156.0000000000, 173.0000000000, 160.0000000000, 49.0000000000, 42.0000000000, 168.0000000000,
56.0000000000, 156.0000000000, 169.0000000000, 164.0000000000, 157.0000000000, 165.0000000000,
163.0000000000, 214.0000000000, 215.0000000000, 170.0000000000, 157.0000000000, 158.0000000000,
164.0000000000, 171.0000000000, 170.0000000000, 159.0000000000, 176.0000000000, 158.0000000000,
172.0000000000, 159.0000000000, 177.0000000000, 171.0000000000, 154.0000000000, 167.0000000000,
160.0000000000, 173.0000000000, 172.0000000000, 178.0000000000, 166.0000000000, 160.0000000000,
161.0000000000, 162.0000000000, 173.0000000000, 156.0000000000, 174.0000000000, 161.0000000000,
179.0000000000, 169.0000000000, 63.0000000000, 56.0000000000, 174.0000000000, 175.0000000000,
168.0000000000, 162.0000000000, 176.0000000000, 215.0000000000, 222.0000000000, 164.0000000000,
227.0000000000, 165.0000000000, 159.0000000000, 165.0000000000, 182.0000000000, 176.0000000000,
166.0000000000, 177.0000000000, 183.0000000000, 178.0000000000, 177.0000000000, 166.0000000000,
167.0000000000, 178.0000000000, 179.0000000000, 167.0000000000, 168.0000000000, 161.0000000000,
184.0000000000, 179.0000000000, 180.0000000000, 168.0000000000, 175.0000000000, 169.0000000000,
185.0000000000, 180.0000000000, 186.0000000000, 174.0000000000, 169.0000000000, 181.0000000000,
70.0000000000, 63.0000000000, 227.0000000000, 165.0000000000, 182.0000000000, 170.0000000000,
171.0000000000, 171.0000000000, 183.0000000000, 182.0000000000, 166.0000000000, 172.0000000000,
188.0000000000, 183.0000000000, 184.0000000000, 172.0000000000, 167.0000000000, 173.0000000000,
168.0000000000, 190.0000000000, 174.0000000000, 184.0000000000, 173.0000000000, 185.0000000000,
175.0000000000, 186.0000000000, 174.0000000000, 185.0000000000, 175.0000000000, 192.0000000000,
186.0000000000, 70.0000000000, 187.0000000000, 171.0000000000, 176.0000000000, 227.0000000000,
237.0000000000, 232.0000000000, 177.0000000000, 188.0000000000, 189.0000000000, 184.0000000000,
188.0000000000, 172.0000000000, 178.0000000000, 177.0000000000, 189.0000000000, 178.0000000000,
179.0000000000, 190.0000000000, 183.0000000000, 173.0000000000, 174.0000000000, 190.0000000000,
179.0000000000, 186.0000000000, 180.0000000000, 191.0000000000, 191.0000000000, 175.0000000000,
192.0000000000, 185.0000000000, 180.0000000000, 181.0000000000, 193.0000000000, 70.0000000000,
74.0000000000, 192.0000000000, 181.0000000000, 189.0000000000, 194.0000000000, 182.0000000000,
237.0000000000, 177.0000000000, 183.0000000000, 242.0000000000, 188.0000000000, 190.0000000000,

195.0000000000, 184.0000000000, 183.0000000000, 194.0000000000, 189.0000000000, 185.0000000000,
200.0000000000, 196.0000000000, 179.0000000000, 195.0000000000, 184.0000000000, 191.0000000000,
196.0000000000, 192.0000000000, 201.0000000000, 197.0000000000, 185.0000000000, 186.0000000000,
190.0000000000, 186.0000000000, 181.0000000000, 193.0000000000, 198.0000000000, 197.0000000000,
191.0000000000, 187.0000000000, 198.0000000000, 187.0000000000, 74.0000000000, 192.0000000000,
199.0000000000, 189.0000000000, 188.0000000000, 242.0000000000, 195.0000000000, 199.0000000000,
200.0000000000, 203.0000000000, 190.0000000000, 194.0000000000, 189.0000000000, 191.0000000000,
190.0000000000, 201.0000000000, 200.0000000000, 198.0000000000, 192.0000000000, 205.0000000000,
201.0000000000, 191.0000000000, 202.0000000000, 202.0000000000, 192.0000000000, 197.0000000000,
242.0000000000, 247.0000000000, 203.0000000000, 252.0000000000, 195.0000000000, 194.0000000000,
195.0000000000, 190.0000000000, 203.0000000000, 204.0000000000, 201.0000000000, 196.0000000000,
207.0000000000, 196.0000000000, 197.0000000000, 191.0000000000, 200.0000000000, 204.0000000000,
205.0000000000, 197.0000000000, 205.0000000000, 206.0000000000, 204.0000000000, 195.0000000000,
200.0000000000, 252.0000000000, 257.0000000000, 199.0000000000, 203.0000000000, 201.0000000000,
200.0000000000, 207.0000000000, 206.0000000000, 207.0000000000, 197.0000000000, 201.0000000000,
204.0000000000, 257.0000000000, 203.0000000000, 207.0000000000, 208.0000000000, 261.0000000000,
206.0000000000, 201.0000000000, 208.0000000000, 204.0000000000, 206.0000000000, 261.0000000000,
211.0000000000, 97.0000000000, 210.0000000000, 211.0000000000, 97.0000000000, 212.0000000000,
213.0000000000, 218.0000000000, 209.0000000000, 216.0000000000, 210.0000000000, 217.0000000000,
212.0000000000, 223.0000000000, 210.0000000000, 217.0000000000, 211.0000000000, 218.0000000000,
224.0000000000, 104.0000000000, 218.0000000000, 210.0000000000, 219.0000000000, 97.0000000000,
163.0000000000, 164.0000000000, 221.0000000000, 215.0000000000, 164.0000000000, 222.0000000000,
214.0000000000, 221.0000000000, 170.0000000000, 217.0000000000, 220.0000000000, 223.0000000000,
221.0000000000, 212.0000000000, 220.0000000000, 216.0000000000, 211.0000000000, 223.0000000000,
224.0000000000, 213.0000000000, 212.0000000000, 210.0000000000, 228.0000000000, 224.0000000000,
225.0000000000, 104.0000000000, 110.0000000000, 229.0000000000, 213.0000000000, 214.0000000000,
221.0000000000, 217.0000000000, 214.0000000000, 223.0000000000, 215.0000000000, 226.0000000000,
217.0000000000, 220.0000000000, 222.0000000000, 170.0000000000, 215.0000000000, 226.0000000000,
227.0000000000, 221.0000000000, 221.0000000000, 228.0000000000, 217.0000000000, 218.0000000000,
212.0000000000, 226.0000000000, 233.0000000000, 228.0000000000, 213.0000000000, 229.0000000000,
219.0000000000, 218.0000000000, 219.0000000000, 229.0000000000, 110.0000000000, 116.0000000000,
230.0000000000, 234.0000000000, 223.0000000000, 228.0000000000, 222.0000000000, 221.0000000000,
231.0000000000, 227.0000000000, 231.0000000000, 232.0000000000, 176.0000000000, 226.0000000000,
222.0000000000, 170.0000000000, 182.0000000000, 224.0000000000, 233.0000000000, 231.0000000000,
226.0000000000, 218.0000000000, 223.0000000000, 236.0000000000, 233.0000000000, 234.0000000000,
219.0000000000, 225.0000000000, 224.0000000000, 235.0000000000, 122.0000000000, 225.0000000000,
234.0000000000, 116.0000000000, 226.0000000000, 236.0000000000, 228.0000000000, 232.0000000000,
227.0000000000, 231.0000000000, 227.0000000000, 182.0000000000, 236.0000000000, 237.0000000000,
234.0000000000, 236.0000000000, 224.0000000000, 238.0000000000, 229.0000000000, 241.0000000000,
228.0000000000, 225.0000000000, 238.0000000000, 230.0000000000, 243.0000000000, 229.0000000000,
235.0000000000, 233.0000000000, 239.0000000000, 239.0000000000, 240.0000000000, 128.0000000000,
122.0000000000, 234.0000000000, 230.0000000000, 231.0000000000, 228.0000000000, 232.0000000000,
237.0000000000, 241.0000000000, 233.0000000000, 182.0000000000, 242.0000000000, 236.0000000000,
232.0000000000, 188.0000000000, 241.0000000000, 243.0000000000, 246.0000000000, 233.0000000000,
234.0000000000, 241.0000000000, 244.0000000000, 248.0000000000, 243.0000000000, 234.0000000000,
235.0000000000, 240.0000000000, 137.0000000000, 249.0000000000, 128.0000000000, 235.0000000000,
133.0000000000, 239.0000000000, 245.0000000000, 244.0000000000, 242.0000000000, 233.0000000000,
236.0000000000, 237.0000000000, 238.0000000000, 246.0000000000, 241.0000000000, 194.0000000000,
199.0000000000, 247.0000000000, 237.0000000000, 251.0000000000, 188.0000000000, 246.0000000000,
234.0000000000, 248.0000000000, 239.0000000000, 251.0000000000, 238.0000000000, 246.0000000000,
248.0000000000, 249.0000000000, 253.0000000000, 240.0000000000, 239.0000000000, 249.0000000000,
137.0000000000, 254.0000000000, 240.0000000000, 250.0000000000, 241.0000000000, 238.0000000000,
251.0000000000, 242.0000000000, 243.0000000000, 251.0000000000, 199.0000000000, 252.0000000000,
242.0000000000, 251.0000000000, 243.0000000000, 239.0000000000, 244.0000000000, 253.0000000000,
244.0000000000, 254.0000000000, 258.0000000000, 253.0000000000, 240.0000000000, 245.0000000000,
137.0000000000, 140.0000000000, 254.0000000000, 245.0000000000, 255.0000000000, 259.0000000000,
253.0000000000, 243.0000000000, 252.0000000000, 256.0000000000, 242.0000000000, 247.0000000000,
248.0000000000, 246.0000000000, 251.0000000000, 203.0000000000, 199.0000000000, 247.0000000000,

256.0000000000, 257.0000000000, 260.0000000000, 260.0000000000, 256.0000000000, 258.0000000000,
244.0000000000, 248.0000000000, 249.0000000000, 251.0000000000, 258.0000000000, 245.0000000000,
259.0000000000, 249.0000000000, 250.0000000000, 140.0000000000, 142.0000000000, 250.0000000000,
260.0000000000, 253.0000000000, 251.0000000000, 252.0000000000, 252.0000000000, 203.0000000000,
263.0000000000, 261.0000000000, 206.0000000000, 260.0000000000, 263.0000000000, 262.0000000000,
249.0000000000, 260.0000000000, 254.0000000000, 253.0000000000, 259.0000000000, 250.0000000000,
255.0000000000, 258.0000000000, 254.0000000000, 256.0000000000, 252.0000000000, 253.0000000000,
258.0000000000, 263.0000000000, 257.0000000000, 263.0000000000, 206.0000000000, 257.0000000000,
258.0000000000, 259.0000000000, 262.0000000000, 260.0000000000, 258.0000000000, 257.0000000000],[0.↩
0000000000, 1.0000000000, 2.0000000000, 3.0000000000, 4.0000000000, 5.0000000000, 6.0000000000,
7.0000000000, 8.0000000000, 9.0000000000, 10.0000000000, 11.0000000000, 12.0000000000, 13.↩
0000000000, 14.0000000000, 15.0000000000, 16.0000000000, 17.0000000000, 18.0000000000, 19.↩
0000000000, 20.0000000000, 21.0000000000, 22.0000000000, 23.0000000000, 24.0000000000, 25.↩
0000000000, 26.0000000000, 27.0000000000, 28.0000000000, 29.0000000000, 30.0000000000, 31.↩
0000000000, 32.0000000000, 33.0000000000, 34.0000000000, 35.0000000000, 36.0000000000, 37.↩
0000000000, 38.0000000000, 39.0000000000, 40.0000000000, 41.0000000000, 42.0000000000, 43.↩
0000000000, 44.0000000000, 45.0000000000, 46.0000000000, 47.0000000000, 48.0000000000, 49.↩
0000000000, 50.0000000000, 51.0000000000, 52.0000000000, 53.0000000000, 54.0000000000, 55.↩
0000000000, 56.0000000000, 57.0000000000, 58.0000000000, 59.0000000000, 60.0000000000, 61.↩
0000000000, 62.0000000000, 63.0000000000, 64.0000000000, 65.0000000000, 66.0000000000, 67.↩
0000000000, 68.0000000000, 69.0000000000, 70.0000000000, 71.0000000000, 72.0000000000, 73.↩
0000000000, 74.0000000000, 75.0000000000, 76.0000000000, 77.0000000000, 78.0000000000, 79.↩
0000000000, 80.0000000000, 81.0000000000, 82.0000000000, 83.0000000000, 84.0000000000, 85.↩
0000000000, 86.0000000000, 87.0000000000, 88.0000000000, 89.0000000000, 90.0000000000, 91.↩
0000000000, 92.0000000000, 93.0000000000, 94.0000000000, 95.0000000000, 96.0000000000, 97.↩
0000000000, 98.0000000000, 99.0000000000, 100.0000000000, 101.0000000000, 102.0000000000,
103.0000000000, 104.0000000000, 105.0000000000, 106.0000000000, 107.0000000000, 108.0000000000,
109.0000000000, 110.0000000000, 111.0000000000, 112.0000000000, 113.0000000000, 114.0000000000,
115.0000000000, 116.0000000000, 117.0000000000, 118.0000000000, 119.0000000000, 120.0000000000,
121.0000000000, 122.0000000000, 123.0000000000, 124.0000000000, 125.0000000000, 126.0000000000,
127.0000000000, 128.0000000000, 129.0000000000, 130.0000000000, 131.0000000000, 132.0000000000,
133.0000000000, 134.0000000000, 135.0000000000, 136.0000000000, 137.0000000000, 138.0000000000,
139.0000000000, 140.0000000000, 141.0000000000, 142.0000000000, 143.0000000000, 144.0000000000,
145.0000000000, 146.0000000000, 147.0000000000, 148.0000000000, 149.0000000000, 150.0000000000,
151.0000000000, 152.0000000000, 153.0000000000, 154.0000000000, 155.0000000000, 156.0000000000,
157.0000000000, 158.0000000000, 159.0000000000, 160.0000000000, 161.0000000000, 162.0000000000,
163.0000000000, 164.0000000000, 165.0000000000, 166.0000000000, 167.0000000000, 168.0000000000,
169.0000000000, 170.0000000000, 171.0000000000, 172.0000000000, 173.0000000000, 174.0000000000,
175.0000000000, 176.0000000000, 177.0000000000, 178.0000000000, 179.0000000000, 180.0000000000,
181.0000000000, 182.0000000000, 183.0000000000, 184.0000000000, 185.0000000000, 186.0000000000,
187.0000000000, 188.0000000000, 189.0000000000, 190.0000000000, 191.0000000000, 192.0000000000,
193.0000000000, 194.0000000000, 195.0000000000, 196.0000000000, 197.0000000000, 198.0000000000,
199.0000000000, 200.0000000000, 201.0000000000, 202.0000000000, 203.0000000000, 204.0000000000,
205.0000000000, 206.0000000000, 207.0000000000, 208.0000000000, 209.0000000000, 210.0000000000,
211.0000000000, 212.0000000000, 213.0000000000, 214.0000000000, 215.0000000000, 216.0000000000,
217.0000000000, 218.0000000000, 219.0000000000, 220.0000000000, 221.0000000000, 222.0000000000,
223.0000000000, 224.0000000000, 225.0000000000, 226.0000000000, 227.0000000000, 228.0000000000,
229.0000000000, 230.0000000000, 231.0000000000, 232.0000000000, 233.0000000000, 234.0000000000,
235.0000000000, 236.0000000000, 237.0000000000, 238.0000000000, 239.0000000000, 240.0000000000,
241.0000000000, 242.0000000000, 243.0000000000, 244.0000000000, 245.0000000000, 246.0000000000,
247.0000000000, 248.0000000000, 249.0000000000, 250.0000000000, 251.0000000000, 252.0000000000,
253.0000000000, 254.0000000000, 255.0000000000, 256.0000000000, 257.0000000000, 258.0000000000,
259.0000000000, 260.0000000000, 261.0000000000, 262.0000000000, 263.0000000000, 0.0000000000,
0.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000,
1.0000000000, 2.0000000000, 2.0000000000, 2.0000000000, 2.0000000000, 3.0000000000, 3.0000000000,
3.0000000000, 3.0000000000, 3.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000,
4.0000000000, 4.0000000000, 5.0000000000, 5.0000000000, 5.0000000000, 6.0000000000, 6.0000000000,
6.0000000000, 6.0000000000, 6.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 7.0000000000,

```
8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000,
9.0000000000, 9.0000000000, 9.0000000000, 9.0000000000, 9.0000000000, 10.0000000000, 10.↩
0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 11.↩
0000000000, 11.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.↩
0000000000, 12.0000000000, 13.0000000000, 13.0000000000, 13.0000000000, 14.0000000000, 14.↩
0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 15.0000000000, 15.↩
0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 16.0000000000, 16.↩
0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 17.0000000000, 17.↩
0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 18.↩
0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.↩
0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.↩
0000000000, 20.0000000000, 20.0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 21.↩
0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 22.0000000000, 22.0000000000, 22.↩
0000000000, 22.0000000000, 22.0000000000, 22.0000000000, 23.0000000000, 23.0000000000, 23.↩
0000000000, 23.0000000000, 23.0000000000, 24.0000000000, 24.0000000000, 24.0000000000, 24.↩
0000000000, 24.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.↩
0000000000, 25.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 26.↩
0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.↩
0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.↩
0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.↩
0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.↩
0000000000, 30.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.↩
0000000000, 31.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.↩
0000000000, 32.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.↩
0000000000, 33.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.↩
0000000000, 34.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.↩
0000000000, 35.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 36.↩
0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.↩
0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.↩
0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.↩
0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.↩
0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.↩
0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.↩
0000000000, 42.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.↩
0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 44.0000000000, 44.0000000000, 44.↩
0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 45.0000000000, 45.↩
0000000000, 45.0000000000, 45.0000000000, 45.0000000000, 46.0000000000, 46.0000000000, 46.↩
0000000000, 46.0000000000, 46.0000000000, 46.0000000000, 47.0000000000, 47.0000000000, 47.↩
0000000000, 47.0000000000, 47.0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 48.↩
0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 49.0000000000, 49.0000000000, 49.↩
0000000000, 49.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 50.↩
0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 52.↩
0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.↩
0000000000, 52.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 53.↩
0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.↩
0000000000, 54.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 55.↩
0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.↩
0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.↩
0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 59.↩
0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 60.↩
0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 61.↩
0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.↩
0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.↩
0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.↩
0000000000, 63.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.↩
0000000000, 64.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.↩
0000000000, 65.0000000000, 65.0000000000, 66.0000000000, 66.0000000000, 67.0000000000, 67.↩
0000000000, 67.0000000000, 67.0000000000, 68.0000000000, 68.0000000000, 68.0000000000, 68.↩
```

```
0000000000, 68.0000000000, 68.0000000000, 69.0000000000, 69.0000000000, 69.0000000000, 70.↩
0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.↩
0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 72.↩
0000000000, 72.0000000000, 72.0000000000, 73.0000000000, 73.0000000000, 74.0000000000, 74.↩
0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 75.0000000000, 75.↩
0000000000, 75.0000000000, 75.0000000000, 76.0000000000, 76.0000000000, 77.0000000000, 77.↩
0000000000, 77.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.↩
0000000000, 78.0000000000, 79.0000000000, 79.0000000000, 79.0000000000, 79.0000000000, 80.↩
0000000000, 80.0000000000, 80.0000000000, 80.0000000000, 81.0000000000, 81.0000000000, 81.↩
0000000000, 81.0000000000, 81.0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 82.↩
0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 83.0000000000, 83.0000000000, 84.↩
0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.↩
0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.↩
0000000000, 86.0000000000, 86.0000000000, 86.0000000000, 86.0000000000, 87.0000000000, 87.↩
0000000000, 87.0000000000, 87.0000000000, 87.0000000000, 88.0000000000, 88.0000000000, 88.↩
0000000000, 88.0000000000, 88.0000000000, 88.0000000000, 89.0000000000, 89.0000000000, 89.↩
0000000000, 89.0000000000, 89.0000000000, 89.0000000000, 90.0000000000, 90.0000000000, 90.↩
0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 91.↩
0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 92.↩
0000000000, 92.0000000000, 92.0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 93.↩
0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 94.0000000000, 94.0000000000, 94.↩
0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 95.0000000000, 95.↩
0000000000, 95.0000000000, 95.0000000000, 95.0000000000, 96.0000000000, 96.0000000000, 96.↩
0000000000, 96.0000000000, 96.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.↩
0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 98.0000000000, 98.↩
0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 100.↩
0000000000, 100.0000000000, 100.0000000000, 100.0000000000, 101.0000000000, 101.0000000000,
101.0000000000, 101.0000000000, 101.0000000000, 101.0000000000, 102.0000000000, 102.0000000000,
102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 103.0000000000,
103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000,
104.0000000000, 104.0000000000, 104.0000000000, 104.0000000000, 104.0000000000, 104.0000000000,
105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000,
105.0000000000, 105.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000,
106.0000000000, 106.0000000000, 106.0000000000, 107.0000000000, 107.0000000000, 107.0000000000,
107.0000000000, 107.0000000000, 108.0000000000, 108.0000000000, 108.0000000000, 108.0000000000,
108.0000000000, 109.0000000000, 109.0000000000, 109.0000000000, 109.0000000000, 109.0000000000,
110.0000000000, 110.0000000000, 110.0000000000, 110.0000000000, 110.0000000000, 111.0000000000,
111.0000000000, 111.0000000000, 111.0000000000, 112.0000000000, 112.0000000000, 112.0000000000,
112.0000000000, 112.0000000000, 112.0000000000, 113.0000000000, 113.0000000000, 113.0000000000,
113.0000000000, 113.0000000000, 113.0000000000, 114.0000000000, 114.0000000000, 114.0000000000,
114.0000000000, 114.0000000000, 114.0000000000, 115.0000000000, 115.0000000000, 115.0000000000,
115.0000000000, 115.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000,
116.0000000000, 116.0000000000, 116.0000000000, 117.0000000000, 117.0000000000, 117.0000000000,
117.0000000000, 117.0000000000, 117.0000000000, 118.0000000000, 118.0000000000, 118.0000000000,
118.0000000000, 118.0000000000, 118.0000000000, 119.0000000000, 119.0000000000, 119.0000000000,
119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 120.0000000000, 120.0000000000,
120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000,
121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000,
121.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000,
122.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000,
123.0000000000, 123.0000000000, 123.0000000000, 124.0000000000, 124.0000000000, 124.0000000000,
124.0000000000, 124.0000000000, 125.0000000000, 125.0000000000, 125.0000000000, 125.0000000000,
126.0000000000, 126.0000000000, 126.0000000000, 126.0000000000, 127.0000000000, 127.0000000000,
127.0000000000, 128.0000000000, 128.0000000000, 128.0000000000, 128.0000000000, 128.0000000000,
129.0000000000, 129.0000000000, 129.0000000000, 129.0000000000, 129.0000000000, 129.0000000000,
130.0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 130.0000000000,
130.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000,
131.0000000000, 131.0000000000, 131.0000000000, 132.0000000000, 132.0000000000, 132.0000000000,
```

132.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 133.0000000000,
134.0000000000, 134.0000000000, 134.0000000000, 134.0000000000, 134.0000000000, 135.0000000000,
135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 136.0000000000,
136.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000,
137.0000000000, 137.0000000000, 137.0000000000, 138.0000000000, 138.0000000000, 138.0000000000,
138.0000000000, 138.0000000000, 139.0000000000, 139.0000000000, 139.0000000000, 140.0000000000,
140.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 141.0000000000,
141.0000000000, 141.0000000000, 141.0000000000, 142.0000000000, 142.0000000000, 143.0000000000,
143.0000000000, 143.0000000000, 143.0000000000, 144.0000000000, 144.0000000000, 144.0000000000,
144.0000000000, 144.0000000000, 145.0000000000, 145.0000000000, 145.0000000000, 146.0000000000,
146.0000000000, 146.0000000000, 146.0000000000, 146.0000000000, 146.0000000000, 147.0000000000,
147.0000000000, 147.0000000000, 147.0000000000, 147.0000000000, 148.0000000000, 148.0000000000,
149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000,
149.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.0000000000,
150.0000000000, 150.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000,
151.0000000000, 151.0000000000, 151.0000000000, 152.0000000000, 152.0000000000, 152.0000000000,
153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000,
154.0000000000, 154.0000000000, 154.0000000000, 154.0000000000, 154.0000000000, 154.0000000000,
155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 156.0000000000,
156.0000000000, 156.0000000000, 156.0000000000, 156.0000000000, 156.0000000000, 157.0000000000,
157.0000000000, 157.0000000000, 157.0000000000, 158.0000000000, 158.0000000000, 158.0000000000,
158.0000000000, 158.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000,
159.0000000000, 159.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 160.0000000000,
160.0000000000, 160.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000,
161.0000000000, 161.0000000000, 162.0000000000, 162.0000000000, 162.0000000000, 162.0000000000,
162.0000000000, 162.0000000000, 163.0000000000, 163.0000000000, 164.0000000000, 164.0000000000,
164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 165.0000000000,
165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 166.0000000000,
166.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.0000000000,
167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000,
168.0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 168.0000000000,
168.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000,
169.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000,
170.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000,
171.0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 172.0000000000,
173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000,
174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000,
175.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 175.0000000000,
175.0000000000, 176.0000000000, 176.0000000000, 176.0000000000, 176.0000000000, 176.0000000000,
177.0000000000, 177.0000000000, 177.0000000000, 177.0000000000, 177.0000000000, 177.0000000000,
178.0000000000, 178.0000000000, 178.0000000000, 178.0000000000, 178.0000000000, 179.0000000000,
179.0000000000, 179.0000000000, 179.0000000000, 179.0000000000, 179.0000000000, 180.0000000000,
180.0000000000, 180.0000000000, 180.0000000000, 181.0000000000, 181.0000000000, 181.0000000000,
181.0000000000, 181.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000,
182.0000000000, 182.0000000000, 182.0000000000, 183.0000000000, 183.0000000000, 183.0000000000,
183.0000000000, 183.0000000000, 183.0000000000, 184.0000000000, 184.0000000000, 184.0000000000,
184.0000000000, 184.0000000000, 184.0000000000, 185.0000000000, 185.0000000000, 185.0000000000,
185.0000000000, 185.0000000000, 185.0000000000, 186.0000000000, 186.0000000000, 186.0000000000,
186.0000000000, 186.0000000000, 186.0000000000, 187.0000000000, 187.0000000000, 187.0000000000,
187.0000000000, 187.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000,
188.0000000000, 188.0000000000, 188.0000000000, 189.0000000000, 189.0000000000, 189.0000000000,
189.0000000000, 189.0000000000, 189.0000000000, 190.0000000000, 190.0000000000, 190.0000000000,
190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 191.0000000000,
191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000,
192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000,
192.0000000000, 193.0000000000, 193.0000000000, 193.0000000000, 193.0000000000, 194.0000000000,
194.0000000000, 194.0000000000, 194.0000000000, 194.0000000000, 195.0000000000, 195.0000000000,
195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 196.0000000000, 196.0000000000,

196.0000000000, 196.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 197.0000000000,
197.0000000000, 197.0000000000, 198.0000000000, 198.0000000000, 198.0000000000, 199.0000000000,
199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 200.0000000000,
200.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 201.0000000000,
201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000,
202.0000000000, 202.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000,
203.0000000000, 203.0000000000, 203.0000000000, 204.0000000000, 204.0000000000, 204.0000000000,
204.0000000000, 204.0000000000, 205.0000000000, 205.0000000000, 205.0000000000, 206.0000000000,
206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 207.0000000000,
207.0000000000, 207.0000000000, 207.0000000000, 208.0000000000, 208.0000000000, 209.0000000000,
209.0000000000, 209.0000000000, 210.0000000000, 210.0000000000, 210.0000000000, 210.0000000000,
210.0000000000, 210.0000000000, 211.0000000000, 211.0000000000, 211.0000000000, 211.0000000000,
212.0000000000, 212.0000000000, 212.0000000000, 212.0000000000, 212.0000000000, 213.0000000000,
213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 214.0000000000,
214.0000000000, 214.0000000000, 214.0000000000, 215.0000000000, 215.0000000000, 215.0000000000,
215.0000000000, 215.0000000000, 216.0000000000, 216.0000000000, 217.0000000000, 217.0000000000,
217.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 218.0000000000, 218.0000000000,
218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 219.0000000000, 219.0000000000,
219.0000000000, 219.0000000000, 219.0000000000, 219.0000000000, 220.0000000000, 220.0000000000,
220.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000,
221.0000000000, 221.0000000000, 222.0000000000, 222.0000000000, 222.0000000000, 222.0000000000,
222.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 223.0000000000,
223.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000,
224.0000000000, 225.0000000000, 225.0000000000, 225.0000000000, 225.0000000000, 225.0000000000,
225.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 226.0000000000,
226.0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 227.0000000000,
227.0000000000, 227.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000,
228.0000000000, 228.0000000000, 228.0000000000, 229.0000000000, 229.0000000000, 229.0000000000,
229.0000000000, 229.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 230.0000000000,
230.0000000000, 231.0000000000, 231.0000000000, 231.0000000000, 231.0000000000, 231.0000000000,
232.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 233.0000000000,
233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000,
234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000,
234.0000000000, 234.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000,
235.0000000000, 235.0000000000, 236.0000000000, 236.0000000000, 236.0000000000, 236.0000000000,
236.0000000000, 236.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000,
237.0000000000, 237.0000000000, 238.0000000000, 238.0000000000, 238.0000000000, 238.0000000000,
238.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000,
239.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000,
240.0000000000, 240.0000000000, 240.0000000000, 241.0000000000, 241.0000000000, 241.0000000000,
241.0000000000, 241.0000000000, 241.0000000000, 242.0000000000, 242.0000000000, 242.0000000000,
242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 243.0000000000,
243.0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 244.0000000000,
244.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 245.0000000000, 245.0000000000,
245.0000000000, 245.0000000000, 245.0000000000, 246.0000000000, 246.0000000000, 246.0000000000,
246.0000000000, 246.0000000000, 247.0000000000, 247.0000000000, 247.0000000000, 247.0000000000,
248.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 249.0000000000,
249.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 250.0000000000,
250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 251.0000000000,
251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000,
251.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000,
252.0000000000, 252.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000,
253.0000000000, 253.0000000000, 253.0000000000, 254.0000000000, 254.0000000000, 254.0000000000,
254.0000000000, 254.0000000000, 255.0000000000, 255.0000000000, 255.0000000000, 256.0000000000,
256.0000000000, 256.0000000000, 256.0000000000, 257.0000000000, 257.0000000000, 257.0000000000,
257.0000000000, 257.0000000000, 257.0000000000, 258.0000000000, 258.0000000000, 258.0000000000,
258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 259.0000000000, 259.0000000000,
259.0000000000, 259.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000,

```
260.0000000000, 260.0000000000, 261.0000000000, 261.0000000000, 261.0000000000, 262.0000000000,
262.0000000000, 263.0000000000, 263.0000000000, 263.0000000000, 263.0000000000]
```

Definition at line 50 of file testCase.py.

6.10.2.8 materialFactor

```
float testCase.materialFactor = specificConductivityMaterial/conducterThickness
```

Definition at line 16 of file testCase.py.

6.10.2.9 Mesh

```
testCase.Mesh = CylindricMeshGiven(meshFile)
```

Definition at line 19 of file testCase.py.

6.10.2.10 meshFile

```
string testCase.meshFile = "cylinder_radius500mm_length1500mm.stl"
```

Definition at line 10 of file testCase.py.

6.10.2.11 reducedSFCorrect

```
list testCase.reducedSFCorrect = [-0.27937980331459932870075136, -0.33966732733873783445233130,
158.30535989173870348167838529, 111.95922396290268352458951995, 58.10903859010657157568857656,
193.39790162602255918500304688, 268.25033740186444219943950884, 189.82215394456304125014867168,
0.25883372957144956671982072, 98.74870633402809971812530421, 215.53653410699894266144838184,
327.39158447388479089568136260, 284.96744613445838467669091187, 202.31101004349304162133194041,
0.82513685120877511280923500, 105.22484885820357192187657347, 223.08256339289710012963041663,
364.76793579171919645887101069, 347.61723438903078431394533254, 186.13386039326010745753592346,
132.63331666423582078095932957, 0.93045442511324694123686641, 69.59377646537200234888587147,
377.49981102452846926098573022, 387.86131916037487599169253372, 228.06934028761415333974582609,
-0.39282069172599420880942489, 0.28340911989108974466944346, 1.32922047073893168089853134,
0.77036064878831922442259383, 402.01980094098689733073115349, 253.64580857732585172925610095,
-0.71137220262248490598722128, -187.02281867595976905249699485, -132.12396908339931655973487068,
1.00935338967148879874002887, -68.31938968731270733769633807, 262.84597566425361492292722687,
-1.01378116441184573659484158, -229.49291759567395843077974860, -285.00916079306904293844127096,
-202.15395986486728929776290897, 0.20369749822588545384860481, -105.12807714412967641237628413,
-1.65128343645850605980740511, -255.65451390973146317264763638, -348.80806087323372821629163809,
-268.84160316836425863584736362, -190.14226619109740568092092872, -0.35372141836276682624884415,
-98.67697413471481127089646179, -265.72733211685635978938080370, -390.53074044891235416798735969,
-328.54385597348704095566063188, -159.10063256399396891538344789, -112.66830824900223717577318894,
```

-0.48136222438574449711268244, -58.492965775446009502047672870, -406.26733909792579879649565555,
-367.00125201198864033358404413, -194.68519835904922388181148563, -0.38739421188725309264100360,
-380.72973951162691719218855724, -216.95201141250419141215388663, -224.64681969190800714386568870,
-57.64885629013803480802380363, -111.76778906363414023417135468, -97.23151838524054824119957630,
-158.57787734712869109898747411, -189.13083157351150020986096933, -103.47360646666479055966192391,
-194.37812613590944010866223834, -268.14533020187656120469910093, -201.00144759920507908645959105,
-67.33160591182809184829238802, -216.97908089605186887638410553, -328.52149241710583282838342711,
-284.82532200213120177068049088, -130.61038743255170402335352264, 1.1740831305907946102706773 7,
-225.01886593214194931533711497, -366.28282561516010673585697077, -347.97810273027931771139265038,
-184.84955361689296182703401428, 1.3557588455271485372577444 7, 68.83865863464234280399978161,
-379.59834765834438030651654117, -389.38131238244437781759188510, -226.65542721376317558679147623,
2.4514269049178736281646706 6, 132.61870525252811603422742337, 104.51756439557146904917317443,
-402.65102785189810674637556076, -252.04129957093991265537624713, 2.66640808738427992352626461,
188.46031263508777442439168226, 202.68330394428298291131795850, 98.04919160612566031431924785,
-262.11224472208607494394527748, 2.80043348997752872264754842, 231.30365585971119912755966652,
286.80511684477886547028901987, 190.23232474250315249264531303, 57.84805143771897917304158909,
2.85708460601647828980276245, 258.69203442119561486833845265, 350.58867061999404768357635476,
268.52259877181154479330871254, 112.10337467749660333993233507, 266.08331708857917874411214143,
390.90976985716883973509538919, 329.01088679331462572008604184, 158.71128445003517981604090892,
404.59245903407168043486308306, 366.35434029559155533206649125, 194.19573855113847571374208201,
379.94898461640434561559231952, 216.17590064203761812677839771, 223.85736931998752652361872606,
215.30222433132459514126821887, 193.26277766072317376710998360, 364.27701325326768255763454363,
157.72987052894848147843731567, 326.94128157610799689791747369, 388.09280021282381767377955839,
111.18402577739000491874321597, 266.54277820804679777211276814, 346.73233678344666941484319977,
252.57711524406573744272463955, 57.016391657150371941042976 69, 187.85218553479762704228051007,
282.67168803322431358537869528, 224.67611712193377115909243003, -3.14326924385573391873549554,
-0.90010570311670168308637585, 96.17069437543644028210110264, 199.33582061541360985756909940,
182.27846100397448481089668348, -4.57611186272201919678082049, -257.99057047583272606061655097,
-1.82557398572480522602745623, 101.69189406672646214246924501, 127.54445573081555664884945145,
-4.49732115287934419711746159, -232.30379071310053973320464138, -392.35336998126877006143331528,
-2.65962064013652721428115910, 64.23306507635152229340746999, -4.17823235146674676343536703,
-190.62136057342786443769000471, -350.97981405658060793939512223, -367.33195935840376478154212236,
-3.92525467655692494872710085, -4.21737374407493348371644970, -136.02000312436450712993973866,
-288.23032697439572302755550481, -329.76997539886201593617442995, -217.04860831737392962040170096,
-4.22506971989652058141473390, -72.29251053978725849447073415, -205.62211432499287866448867135,
-270.03888658781465892388951033, -195.10425396920604157458001282, -3.75470502920668458557429403,
-107.71214521643315720211830921, -191.64672875892048864443495404, -159.55913514871488700919144321,
-2.82958247867568246647351771, -100.33839131188909732372849248, -113.28188975877471023068210343,
-2.02984471265764909730933141, -59.20137760431371987124293810, -1.13006153433920530915202107,
-216.94986764694439784761925694, -194.17815277072924118328955956, -366.46414265432213142048567533,
-58.68817436396830089506693184, -158.58371851794368012633640319, -328.49641010394003615147084929,
-389.37906114078765540398308076, -112.59894550018677250591281336, -99.47273320793448192489449866,
-268.50544005644428580126259476, -349.78532743252520731402910315, -254.30643984757679731956159230,
-190.57684701519576719874748960, -106.72241271551203567469201516, -286.77766766087427185993874446,
-228.97129601689493938465602696, 0.65888959003758174048925866, -203.43166351223138121895317454,
-71.95880175563581815367797390, -188.94812353702479867934016511, -0.54028097490086324228286685,
255.71015351091924117099551950, -135.09478864364308492440613918, -4.18373247132461756336851977,
-1.97138030481194448384485440, 228.04006865312763352449110243, 391.09379142082968883187277243,
-3.47586424690169337736733723, 65.01418211509854927498963661, 184.65606105019628557784017175,
348.36366144437067760009085760, 366.20261451786950601672288030, 128.93974257358527779615542386,
101.83529783301392512839811388, 283.51551025120562599113327451, 327.60501736958264018539921381,
216.03661549146309539537469391, 200.00439085138680184172699228, 96.43013782801264710542454850,
267.22322299296490655251545832, 193.49255409229277802296564914, 188.08418865096339800402347464,
57.04436608707574407617357792, 157.96464661316298361271037720, 111.31922987220724508006242104]

Definition at line 56 of file testCase.py.

6.10.2.12 resistanceMatrix

```
testCase.resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
```

Definition at line 22 of file testCase.py.

6.10.2.13 sensitivityMatrix

```
testCase.sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
```

Definition at line 21 of file testCase.py.

6.10.2.14 SFCorrectValue

```
list testCase.SFCorrectValue = [-0.27937980331459933, 158.30535989173870348, -0.27937980331459933,
111.95922396290268352, 58.10903859010657158, -0.27937980331459933, 193.39790162602255919,
-0.27937980331459933, 268.25033740186444220, 189.82215394456304125, 0.25883372957144957, -0.↩
27937980331459933, 98.74870633402809972, -0.27937980331459933, 215.53653410699894266, 327.↩
39158447388479090, 284.96744613445838468, 202.31101004349304162, 0.82513685120877511, 105.↩
22484885820357192, -0.27937980331459933, 223.08256339289710013, 364.76793579171919646, 347.↩
61723438903078431, 186.13386039326010746, 132.63331666423582078, 0.93045442511324694, 69.↩
59377646537200235, 377.49981102452846926, 387.86131916037487599, 228.06934028761415334, -0.↩
39282069172599421, 0.28340911989108974, 1.32922047073893168, 0.77036064878831922, 402.01980094098689733,
253.64580857732585173, -0.71137220262248491, -187.02281867595976905, -132.12396908339931656,
1.00935338967148880, -68.31938968731270734, 262.84597566425361492, -1.01378116441184574, -229.↩
49291759567395843, -285.00916079306904294, -202.15395986486728930, 0.20369749822588545, -105.↩
12807714412967641, -1.65128343645850606, -255.65451390973146317, -348.80806087323372822, -268.↩
84160316836425864, -190.14226619109740568, -0.35372141836276683, -98.67697413471481127, -265.↩
72733211685635979, -390.53074044891235417, -328.54385597348704096, -159.10063256399396892,
-112.66830824900223718, -0.48136222438574450, -58.49296577544600950, -406.26733909792579880,
-367.00125201198864033, -194.68519835904922388, -0.33966732733873783, -0.33966732733873783,
-0.38739421188725309, -0.33966732733873783, -380.72973951162691719, -216.95201141250419141,
-0.33966732733873783, -0.33966732733873783, -224.64681969190800714, -0.33966732733873783,
-0.33966732733873783, -0.27937980331459933, -57.64885629013803481, -0.27937980331459933, -111.↩
76778906363414023, -97.23151838524054824, -158.57787734712869110, -0.27937980331459933, -189.↩
13083157351150021, -103.47360646666479056, -0.27937980331459933, -194.37812613590944011, -268.↩
14533020187656120, -201.00144759920507909, -67.33160591182809185, -216.97908089605186888,
-0.27937980331459933, -328.52149241710583283, -284.82532200213120177, -130.61038743255170402,
1.17408313059079461, -225.01886593214194932, -0.27937980331459933, -366.28282561516010674,
-347.97810273027931771, -184.84955361689296183, 1.35575884552714854, 68.83865863464234280,
-379.59834765834438031, -389.38131238244437782, -226.65542721376317559, 2.45142690491787363,
132.61870525252811603, 104.51756439557146905, -402.65102785189810675, -252.04129957093991266,
2.66640808738427992, 188.46031263508777442, 202.68330394428298291, 98.04919160612566031, -262.↩
11224472208607494, 2.80043348997752872, 231.30365585971119913, 286.80511684477886547, 190.↩
23232474250315249, 57.84805143771897917, 2.85708460601647829, 258.69203442119561487, 350.↩
58867061999404768, 268.52259877181154479, 112.10337467749660334, -0.33966732733873783, 266.↩
08331708857917874, 390.90976985716883974, 329.01088679331462572, 158.71128445003517982, -0.↩
33966732733873783, 404.59245903407168043, 366.35434029559155533, 194.19573855113847571, -0.↩
33966732733873783, 379.94898461640434562, 216.17590064203761813, -0.33966732733873783, 223.↩
85736931998752652, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, 215.↩
```

```
30222433132459514, -0.27937980331459933, 193.26277766072317377, 364.27701325326768256, -0.↩
27937980331459933, 157.72987052894848148, 326.94128157610799690, 388.09280021282381767, -0.↩
27937980331459933, 111.18402577739000492, 266.54277820804679777, 346.73233678344666941, 252.↩
57711524406573744, -0.27937980331459933, 57.01639165715037194, 187.85218553479762704, 282.↩
67168803322431359, 224.67611712193377116, -3.14326924385573392, -0.27937980331459933, -0.↩
90010570311670168, 96.17069437543644028, 199.33582061541360986, 182.27846100397448481, -4.↩
57611186272201920, -257.99057047583272606, -1.82557398572480523, 101.69189406672646214, 127.↩
54445573081555665, -4.49732115287934420, -232.30379071310053973, -392.35336998126877006, -2.↩
65962064013652721, 64.23306507635152229, -4.17823235146674676, -190.62136057342786444, -350.↩
97981405658060794, -367.33195935840376478, -3.92525467655692495, -4.21737374407493348, -136.↩
02000312436450713, -288.23032697439572303, -329.76997539886201594, -217.04860831373792962,
-4.22506971989652058, -72.29251053978725849, -205.62211432499287866, -270.03888658781465892,
-195.10425396920604157, -0.33966732733873783, -3.75470502920668459, -107.71214521643315720,
-191.64672875892048864, -159.55913514871488701, -0.33966732733873783, -2.82958247867568247,
-100.33839131188909732, -113.28188975877471023, -0.33966732733873783, -2.02984471265764910,
-59.20137760431371987, -0.33966732733873783, -1.13006153433920531, -0.33966732733873783, -0.↩
33966732733873783, -0.27937980331459933, -216.94986764694439785, -0.27937980331459933, -194.↩
17815277072924118, -366.46414265432213142, -0.27937980331459933, -58.68817436396830090, -0.↩
27937980331459933, -158.58371851794368013, -328.49641010394003615, -389.37906114078765540,
-0.27937980331459933, -112.59894550018677251, -99.47273320793448192, -268.50544005644428580,
-349.78532743252520731, -254.30643984757679732, -190.57684701519576720, -106.72241271551203567,
-286.77766766087427186, -228.97129601689493938, 0.65888959003758174, -203.43166351223138122,
-71.95880175563581815, -188.94812353702479868, -0.54028097490086324, 255.71015351091924117,
-135.09478864364308492, -4.18373247132461756, -1.97138030481194448, 228.04006865312763352,
391.09379142082968883, -3.47586424690169338, 65.01418211509854927, 184.65606105019628558,
348.36366144437067760, 366.20261451786950602, 128.93974257358527780, 101.83529783301392513,
283.51551025120562599, 327.60501736958264019, 216.03661549146309540, 200.00439085138680184,
96.43013782801264711, 267.22322299296490655, 193.49255409229277802, -0.33966732733873783,
188.08418865096339800, 57.04436608707574408, 157.96464661316298361, -0.33966732733873783,
111.31922987220724508, -0.33966732733873783, -0.33966732733873783, -0.33966732733873783]
```

Definition at line 58 of file testCase.py.

6.10.2.15 SFOpt

```
testCase.SFOpt
```

Definition at line 23 of file testCase.py.

6.10.2.16 specificConductivityMaterial

```
float testCase.specificConductivityMaterial = 1.8000*10**-8
```

Definition at line 14 of file testCase.py.

6.10.2.17 targetMeshFile

```
string testCase.targetMeshFile = "sphere_radius150mm.stl"
```

Definition at line 11 of file testCase.py.

6.10.2.18 TargetSphere

```
testCase.TargetSphere = TargetFieldGiven(targetMeshFile,1)
```

Definition at line 20 of file testCase.py.

6.10.2.19 Test

```
testCase.Test = Tester()
```

Definition at line 18 of file testCase.py.

6.10.2.20 tikonovFac

```
int testCase.tikonovFac = 100
```

Definition at line 13 of file testCase.py.

6.11 Tester Namespace Reference

Classes

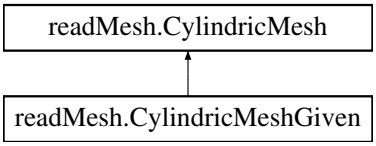
- class Tester

Chapter 7

Class Documentation

7.1 readMesh.CylindricMesh Class Reference

Inheritance diagram for readMesh.CylindricMesh:



Public Member Functions

- def __init__ (self, coilLength, coilRadius, n)
- def checkIfBoundary (self)
- def getVertexNormals (self)
- def getNormals (self)
- def getNeighbourAreas (self)
- def getNeighbourCurrents (self)
- def getNeighbourCurrentsUnsorted (self)
- def getCurrent (self)
- def getAreas (self)
- def getOpenBoundaries (self)
- def getBoundaryEdges (self)
- def removeDoubleEdges (self, edgeList)
- def getBoundaryLoopNodes (self)
- def getRotatedCopy (self)
- def getRotatedVertices (self, rotMat)
- def calcRotationVec (self, boundaryLoopNodes)
- def calc3DRotMatByVec (self, rotationVec, angle)
- def turnAnsSortElements (self, boundaryEdges)
- def get2Dcoordinates (self)
- def getNeighbourTriangleIndices (self)
- def getOneRingList (self)
- def orderElementsInCircularArangement (self, oneRingList)
- def arrangeCircular (self, start, Elements)
- def findStartInBoundaryCase (self, oneRingList, nodeNumber)
- def checkStartTriangle (self, verticeTriangles, start)
- def createOneRingList (self)
- def ensureUniformOrientation (self, oneRingList)

Public Attributes

- faces
- vertices
- normals
- openBoundaries
- areas
- current
- neighbours
- rotatedCaylinder
- v
- neighbourareas
- currentDensityFaces
- vertexNormals
- boundary
- oneRingList
- neighbourcurrents
- neighbourcurrentUnsorted

7.1.1 Detailed Description

Definition at line 22 of file readMesh.py.

7.1.2 Constructor & Destructor Documentation

7.1.2.1 __init__()

```
def readMesh.CylindricMesh.__init__ (
              self,
              coilLength,
              coilRadius,
              n )
```

Reimplemented in readMesh.CylindricMeshGiven.

Definition at line 23 of file readMesh.py.

7.1.3 Member Function Documentation

7.1.3.1 arrangeCircular()

```
def readMesh.CylindricMesh.arrangeCircular (
              self,
              start,
              Elements )
```

```
returns the elements in a circular order beginning with start
```

Definition at line 273 of file readMesh.py.

7.1.3.2 calc3DRotMatByVec()

```
def readMesh.CylindricMesh.calc3DRotMatByVec (
              self,
              rotationVec,
              angle )
```

```
returns the rotation matrix calculated from the rotationVector
```

Definition at line 191 of file readMesh.py.

7.1.3.3 calcRotationVec()

```
def readMesh.CylindricMesh.calcRotationVec (
              self,
              boundaryLoopNodes )
```

```
returns the rotationVector and the angle based on the boundaryLoopNodes.
```

Definition at line 179 of file readMesh.py.

7.1.3.4 checkIfBoundary()

```
def readMesh.CylindricMesh.checkIfBoundary (
              self )
```

```
returns a list of boolean if the vertice is a boundary vertice
```

Definition at line 41 of file readMesh.py.

7.1.3.5 checkStartTriangle()

```
def readMesh.CylindricMesh.checkStartTriangle (
              self,
              verticeTriangles,
              start )
```

returns boolean if "start" is the correct startTriangle

Definition at line 295 of file readMesh.py.

7.1.3.6 createOneRingList()

```
def readMesh.CylindricMesh.createOneRingList (
              self )
```

returns a list with the other two triangle Points for each triangle per node

Definition at line 302 of file readMesh.py.

7.1.3.7 ensureUniformOrientation()

```
def readMesh.CylindricMesh.ensureUniformOrientation (
              self,
              oneRingList )
```

returns oneRingList with ensured uniform Orientation

Definition at line 317 of file readMesh.py.

7.1.3.8 findStartInBoundaryCase()

```
def readMesh.CylindricMesh.findStartInBoundaryCase (
              self,
              oneRingList,
              nodeNumber )
```

returns the correct start triangle for ordering the triangles around a boundary vertice

Definition at line 284 of file readMesh.py.

7.1.3.9 get2Dcoordinates()

```
def readMesh.CylindricMesh.get2Dcoordinates (
              self )
```

returns the from 3D to 2D converted vertices

Definition at line 231 of file readMesh.py.

7.1.3.10 getAreas()

```
def readMesh.CylindricMesh.getAreas (
              self )
```

returns the areas of the triangles made with the points in faces

Definition at line 103 of file readMesh.py.

7.1.3.11 getBoundaryEdges()

```
def readMesh.CylindricMesh.getBoundaryEdges (
              self )
```

returns the nodes for each boundary in the correct order

Definition at line 130 of file readMesh.py.

7.1.3.12 getBoundaryLoopNodes()

```
def readMesh.CylindricMesh.getBoundaryLoopNodes (
              self )
```

returns the unsorted nodes for the boundaryLoop.

Definition at line 153 of file readMesh.py.

7.1.3.13 getCurrent()

```
def readMesh.CylindricMesh.getCurrent (
              self )
```

returns the current for the triangles made with the points in faces
C = (c-b)/(2*Fläche)

Definition at line 95 of file readMesh.py.

7.1.3.14 getNeighbourAreas()

```
def readMesh.CylindricMesh.getNeighbourAreas (
              self )
```

returns the areas of the neighbour triangles for every node

Definition at line 65 of file readMesh.py.

7.1.3.15 getNeighbourCurrents()

```
def readMesh.CylindricMesh.getNeighbourCurrents (
              self )
```

returns the currents of the neighbour triangles for every node

Definition at line 75 of file readMesh.py.

7.1.3.16 getNeighbourCurrentsUnsorted()

```
def readMesh.CylindricMesh.getNeighbourCurrentsUnsorted (
              self )
```

returns the currents of the neighbour triangles for every node before sorting

Definition at line 85 of file readMesh.py.

7.1.3.17 getNeighbourTriangleIndices()

```
def readMesh.CylindricMesh.getNeighbourTriangleIndices (
              self )
```

```
returns the indices of the neighbour triangles of every node
```

Definition at line 242 of file readMesh.py.

7.1.3.18 getNormals()

```
def readMesh.CylindricMesh.getNormals (
              self )
```

```
returns the normals of the faces
```

Definition at line 58 of file readMesh.py.

7.1.3.19 getOneRingList()

```
def readMesh.CylindricMesh.getOneRingList (
              self )
```

```
returns sorted list with nodes around every node
```

Definition at line 254 of file readMesh.py.

7.1.3.20 getOpenBoundaries()

```
def readMesh.CylindricMesh.getOpenBoundaries (
              self )
```

```
returns indexes of the nodes at the edges of a cylinder extended in z-direction
```

Definition at line 110 of file readMesh.py.

7.1.3.21 getRotatedCopy()

```
def readMesh.CylindricMesh.getRotatedCopy (
              self )
```

returns rotated copy of the vertices. If the cylinder is orientated along the z axis we need a rotated copy.

Definition at line 164 of file readMesh.py.

7.1.3.22 getRotatedVertices()

```
def readMesh.CylindricMesh.getRotatedVertices (
              self,
              rotMat )
```

returns the rotated vertices (multiplication with rotMat).

Definition at line 172 of file readMesh.py.

7.1.3.23 getVertexNormals()

```
def readMesh.CylindricMesh.getVertexNormals (
              self )
```

returns the normals of the vertices. These are calculated as average of the touching faces normals.

Definition at line 48 of file readMesh.py.

7.1.3.24 orderElementsInCircularArangement()

```
def readMesh.CylindricMesh.orderElementsInCircularArangement (
              self,
              oneRingList )
```

returns the List in a circular arrangement

Definition at line 261 of file readMesh.py.

7.1.3.25 removeDoubleEdges()

```
def readMesh.CylindricMesh.removeDoubleEdges (
              self,
              edgeList )
```

```
returns the edgeList with each edge just once.
```

Definition at line 144 of file readMesh.py.

7.1.3.26 turnAnsSortElements()

```
def readMesh.CylindricMesh.turnAnsSortElements (
              self,
              boundaryEdges )
```

```
returns the given list in sorted. If needed single elements were turned to close the loop.
```

Definition at line 209 of file readMesh.py.

7.1.4 Member Data Documentation

7.1.4.1 areas

```
readMesh.CylindricMesh.areas
```

Definition at line 28 of file readMesh.py.

7.1.4.2 boundary

```
readMesh.CylindricMesh.boundary
```

Definition at line 36 of file readMesh.py.

7.1.4.3 current

```
readMesh.CylindricMesh.current
```

Definition at line 29 of file readMesh.py.

7.1.4.4 currentDensityFaces

`readMesh.CylindricMesh.currentDensityFaces`

Definition at line 34 of file readMesh.py.

7.1.4.5 faces

`readMesh.CylindricMesh.faces`

Definition at line 24 of file readMesh.py.

7.1.4.6 neighbourareas

`readMesh.CylindricMesh.neighbourareas`

Definition at line 33 of file readMesh.py.

7.1.4.7 neighbourcurrents

`readMesh.CylindricMesh.neighbourcurrents`

Definition at line 38 of file readMesh.py.

7.1.4.8 neighbourcurrentUnsorted

`readMesh.CylindricMesh.neighbourcurrentUnsorted`

Definition at line 39 of file readMesh.py.

7.1.4.9 neighbours

`readMesh.CylindricMesh.neighbours`

Definition at line 30 of file readMesh.py.

7.1.4.10 normals

`readMesh.CylindricMesh.normals`

Definition at line 26 of file readMesh.py.

7.1.4.11 oneRingList

`readMesh.CylindricMesh.oneRingList`

Definition at line 37 of file readMesh.py.

7.1.4.12 openBoundaries

`readMesh.CylindricMesh.openBoundaries`

Definition at line 27 of file readMesh.py.

7.1.4.13 rotatedCaylinder

`readMesh.CylindricMesh.rotatedCaylinder`

Definition at line 31 of file readMesh.py.

7.1.4.14 v

`readMesh.CylindricMesh.v`

Definition at line 32 of file readMesh.py.

7.1.4.15 vertexNormals

`readMesh.CylindricMesh.vertexNormals`

Definition at line 35 of file readMesh.py.

7.1.4.16 vertices

`readMesh.CylindricMesh.vertices`

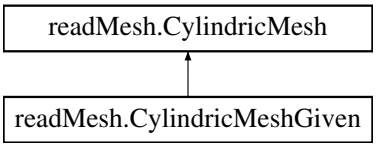
Definition at line 25 of file readMesh.py.

The documentation for this class was generated from the following file:

- subfunctions/readMesh.py

7.2 readMesh.CylindricMeshGiven Class Reference

Inheritance diagram for readMesh.CylindricMeshGiven:



Public Member Functions

- def __init__ (self, filename)

Public Attributes

- faces
- vertices
- normals
- openBoundaries
- areas
- current
- neighbours
- rotatedCaylinder
- v
- neighbourareas
- currentDensityFaces
- vertexNormals
- boundary
- oneRingList
- neighbourcurrents
- neighbourcurrentUnsorted

7.2.1 Detailed Description

Definition at line 340 of file readMesh.py.

7.2.2 Constructor & Destructor Documentation

7.2.2.1 __init__()

```
def readMesh.CylindricMeshGiven.__init__ (
              self,
              filename )
```

Reimplemented from readMesh.CylindricMesh.

Definition at line 341 of file readMesh.py.

7.2.3 Member Data Documentation

7.2.3.1 areas

```
readMesh.CylindricMeshGiven.areas
```

Definition at line 346 of file readMesh.py.

7.2.3.2 boundary

```
readMesh.CylindricMeshGiven.boundary
```

Definition at line 354 of file readMesh.py.

7.2.3.3 current

```
readMesh.CylindricMeshGiven.current
```

Definition at line 347 of file readMesh.py.

7.2.3.4 currentDensityFaces

```
readMesh.CylindricMeshGiven.currentDensityFaces
```

Definition at line 352 of file readMesh.py.

7.2.3.5 faces

`readMesh.CylindricMeshGiven.faces`

Definition at line 342 of file readMesh.py.

7.2.3.6 neighbourareas

`readMesh.CylindricMeshGiven.neighbourareas`

Definition at line 351 of file readMesh.py.

7.2.3.7 neighbourcurrents

`readMesh.CylindricMeshGiven.neighbourcurrents`

Definition at line 356 of file readMesh.py.

7.2.3.8 neighbourcurrentUnsorted

`readMesh.CylindricMeshGiven.neighbourcurrentUnsorted`

Definition at line 357 of file readMesh.py.

7.2.3.9 neighbours

`readMesh.CylindricMeshGiven.neighbours`

Definition at line 348 of file readMesh.py.

7.2.3.10 normals

`readMesh.CylindricMeshGiven.normals`

Definition at line 344 of file readMesh.py.

7.2.3.11 oneRingList

`readMesh.CylindricMeshGiven.oneRingList`

Definition at line 355 of file readMesh.py.

7.2.3.12 openBoundaries

`readMesh.CylindricMeshGiven.openBoundaries`

Definition at line 345 of file readMesh.py.

7.2.3.13 rotatedCaylinder

`readMesh.CylindricMeshGiven.rotatedCaylinder`

Definition at line 349 of file readMesh.py.

7.2.3.14 v

`readMesh.CylindricMeshGiven.v`

Definition at line 350 of file readMesh.py.

7.2.3.15 vertexNormals

`readMesh.CylindricMeshGiven.vertexNormals`

Definition at line 353 of file readMesh.py.

7.2.3.16 vertices

`readMesh.CylindricMeshGiven.vertices`

Definition at line 343 of file readMesh.py.

The documentation for this class was generated from the following file:

- subfunctions/readMesh.py

7.3 showSTLMesh.CylindricMeshGiven Class Reference

Public Member Functions

- def __init__ (self)

Public Attributes

- vertices

7.3.1 Detailed Description

Definition at line 12 of file showSTLMesh.py.

7.3.2 Constructor & Destructor Documentation

7.3.2.1 __init__()

```
def showSTLMesh.CylindricMeshGiven.__init__ (
              self )
```

Definition at line 13 of file showSTLMesh.py.

7.3.3 Member Data Documentation

7.3.3.1 vertices

```
showSTLMesh.CylindricMeshGiven.vertices
```

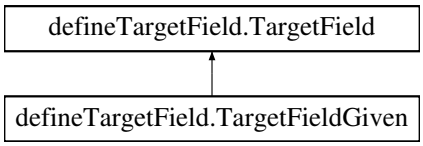
Definition at line 14 of file showSTLMesh.py.

The documentation for this class was generated from the following file:

- showSTLMesh.py

7.4 defineTargetField.TargetField Class Reference

Inheritance diagram for defineTargetField.TargetField:



Public Member Functions

- def __init__ (self, center, radius, direction)
- def getTargetPoints (self)
- def getMagneticFieldValues (self, direction)

Public Attributes

- center
- radius
- vertices
- fieldValues

7.4.1 Detailed Description

Definition at line 8 of file defineTargetField.py.

7.4.2 Constructor & Destructor Documentation

7.4.2.1 __init__()

```
def defineTargetField.TargetField.__init__ (
              self,
              center,
              radius,
              direction )
```

Reimplemented in defineTargetField.TargetFieldGiven.

Definition at line 9 of file defineTargetField.py.

7.4.3 Member Function Documentation

7.4.3.1 getMagneticFieldValues()

```
def defineTargetField.TargetField.getMagneticFieldValues (
              self,
              direction )
```

```
returns magnetic field values analogus to matlab skript
```

Definition at line 30 of file defineTargetField.py.

7.4.3.2 getTargetPoints()

```
def defineTargetField.TargetField.getTargetPoints (
              self )
```

```
generate Target Points within a circle with the given specifications
```

Definition at line 15 of file defineTargetField.py.

7.4.4 Member Data Documentation

7.4.4.1 center

```
defineTargetField.TargetField.center
```

Definition at line 10 of file defineTargetField.py.

7.4.4.2 fieldValues

```
defineTargetField.TargetField.fieldValues
```

Definition at line 13 of file defineTargetField.py.

7.4.4.3 radius

```
defineTargetField.TargetField.radius
```

Definition at line 11 of file defineTargetField.py.

7.4.4.4 vertices

```
defineTargetField.TargetField.vertices
```

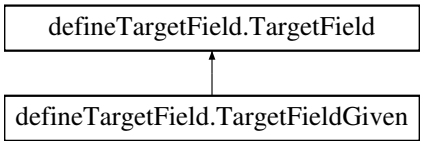
Definition at line 12 of file defineTargetField.py.

The documentation for this class was generated from the following file:

- subfunctions/defineTargetField.py

7.5 defineTargetField.TargetFieldGiven Class Reference

Inheritance diagram for defineTargetField.TargetFieldGiven:



Public Member Functions

- def __init__ (self, filename, direction)

Public Attributes

- faces
- fieldValues

7.5.1 Detailed Description

Definition at line 38 of file defineTargetField.py.

7.5.2 Constructor & Destructor Documentation

7.5.2.1 __init__()

```
def defineTargetField.TargetFieldGiven.__init__ (
              self,
              filename,
              direction )
```

Reimplemented from defineTargetField.TargetField.

Definition at line 39 of file defineTargetField.py.

7.5.3 Member Data Documentation

7.5.3.1 faces

`defineTargetField.TargetFieldGiven.faces`

Definition at line 40 of file defineTargetField.py.

7.5.3.2 fieldValues

`defineTargetField.TargetFieldGiven.fieldValues`

Definition at line 41 of file defineTargetField.py.

The documentation for this class was generated from the following file:

- subfunctions/defineTargetField.py

7.6 Tester.Tester Class Reference

Public Member Functions

- def __init__ (self)

Public Attributes

- gaußLegendre
- reducedSF
- calcWeightsGauss
- matElementsShouldGetValue

7.6.1 Detailed Description

Definition at line 1 of file Tester.py.

7.6.2 Constructor & Destructor Documentation

7.6.2.1 __init__()

```
def Tester.Tester.__init__ (
              self )
```

Definition at line 2 of file Tester.py.

7.6.3 Member Data Documentation

7.6.3.1 calcWeightsGauss

```
Tester.Tester.calcWeightsGauss
```

Definition at line 5 of file Tester.py.

7.6.3.2 gaußLegendre

```
Tester.Tester.gaußLegendre
```

Definition at line 3 of file Tester.py.

7.6.3.3 matElementsShouldGetValue

```
Tester.Tester.matElementsShouldGetValue
```

Definition at line 6 of file Tester.py.

7.6.3.4 reducedSF

```
Tester.Tester.reducedSF
```

Definition at line 4 of file Tester.py.

The documentation for this class was generated from the following file:

- subfunctions/Tester.py

Chapter 8

File Documentation

8.1 main.py File Reference

Namespaces

- namespace main

Variables

- main.Test = Tester()
- string main.meshFile = "cylinder_radius500mm_length1500mm.stl"

  Input ##################.
- string main.targetMeshFile = "sphere_radius150mm.stl"
- int main.gaussOrder = 2
- int main.tikonovFac = 100
- float main.specificConductivityMaterial = 1.8000∗10∗∗-8
- float main.conducterThickness = 0.005
- float main.materialFactor = specificConductivityMaterial/conducterThickness
- int main.numLevels = 20
- float main.levelOffset = 0.2500
- main.Mesh = CylindricMeshGiven(meshFile)
- main.TargetSphere = TargetFieldGiven(targetMeshFile,1)
- main.sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
- main.resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
- main.bFieldGeneratedByOptSF
- main.streamFunction
- main.contourStep
- main.potentialLevelList
- main.contour = calcContoursByTriangluarPotentialCuts(Mesh,potentialLevelList,streamFunction)

8.2 main.py

Go to the documentation of this file.

```
00001 import numpy as np
00002 from decimal import *
00003 import sys
00004 sys.path.append('subfunctions/')
00005
00006
00007
00008 from subfunctions.Tester import Tester
00009 Test = Tester()
00010
00011
00012 meshFile = "cylinder_radius500mm_length1500mm.stl" #insert Filename of stl mesh or False here
00013 targetMeshFile = "sphere_radius150mm.stl" #insert Filename of stl mesh or False here
00014 gaussOrder = 2
00015 tikonovFac = 100
00016 specificConductivityMaterial = 1.8000*10**-8
00017 conducterThickness = 0.005
00018 materialFactor = specificConductivityMaterial/conducterThickness
00019 numLevels = 20
00020 levelOffset = 0.2500
00021
00022
00023
00024 from subfunctions.readMesh import CylindricMesh,CylindricMeshGiven
00025 if meshFile:
00026     Mesh = CylindricMeshGiven(meshFile)
00027 else: Mesh = CylindricMesh(5.0,3.0,10)
00028
00029 from subfunctions.defineTargetField import TargetField,TargetFieldGiven
00030 if targetMeshFile:
00031     TargetSphere = TargetFieldGiven(targetMeshFile,1)
00032 else: TargetSphere = TargetField([0,0,0],4,1)
00033
00034 from subfunctions.sensitivityMatrix import getSensitivityMatrix
00035 sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
00036
00037 from subfunctions.resistanceMatrix import getResistanceMatrix
00038 resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
00039
00040
00041
00042 from subfunctions.streamFunctionOptimization import streamFunctionOptimization
00043 bFieldGeneratedByOptSF,streamFunction =
       streamFunctionOptimization(Test,Mesh,TargetSphere,sensitivityMatrix,resistanceMatrix,tikonovFac)
00044
00045 from subfunctions.calcPotentialLevels import calcPotentialLevels
00046 contourStep, potentialLevelList = calcPotentialLevels(streamFunction, numLevels, levelOffset)
00047
00048 from subfunctions.calcContoursByTriangularPotentialCuts import calcContoursByTriangluarPotentialCuts
00049 contour = calcContoursByTriangluarPotentialCuts(Mesh,potentialLevelList,streamFunction)
00050
00051 # topological contour sorting
00052
00053 # opening and interconnection wires
00054
00055
00056
00057 # plots
00058
00059 # ouput for 3D
```

8.3 README.md File Reference

8.4 showSTLMesh.py File Reference

Classes

- class showSTLMesh.CylindricMeshGiven

Namespaces

- namespace showSTLMesh

Functions

- def showSTLMesh.getMesh (filename)

Variables

- showSTLMesh.givenMesh = CylindricMeshGiven()
- showSTLMesh.figure = pyplot.figure()
- showSTLMesh.axes = mplot3d.Axes3D(figure)
- showSTLMesh.your_mesh = mesh.Mesh.from_file('cylinder_radius500mm_length1500mm.stl')
- showSTLMesh.scale = your_mesh.points.flatten()

8.5 showSTLMesh.py

Go to the documentation of this file.

```
00001 import numpy as np
00002 from stl import mesh
00003 from mpl_toolkits import mplot3d
00004 from matplotlib import pyplot
00005
00006 def getMesh(filename):
00007     your_mesh = mesh.Mesh.from_file(filename)
00008     normals = your_mesh.normals
00009     vertices = [your_mesh.v0, your_mesh.v1, your_mesh.v2]
00010     return normals, vertices
00011
00012 class CylindricMeshGiven():
00013     def __init__(self):
00014         self.normals, self.vertices = getMesh('cylinder_radius500mm_length1500mm.stl')
00015
00016
00017 givenMesh = CylindricMeshGiven()
00018 print("mesh",givenMesh.normals)
00019
00020
00021 # Create a new plot
00022 figure = pyplot.figure()
00023 axes = mplot3d.Axes3D(figure)
00024
00025 # Using an existing stl file:
00026
00027
00028 your_mesh = mesh.Mesh.from_file('cylinder_radius500mm_length1500mm.stl')
00029
00030 print(your_mesh.normals)
00031 axes.add_collection3d(mplot3d.art3d.Poly3DCollection(your_mesh.vectors))
00032
00033
00034 # Auto scale to the mesh size
00035 scale = your_mesh.points.flatten()
00036 axes.auto_scale_xyz(scale, scale, scale)
00037
00038 # Show the plot to the screen
00039 pyplot.show()
```

8.6 subfunctions/calcContoursByTriangularPotentialCuts.py File Reference

Namespaces

- namespace calcContoursByTriangularPotentialCuts

Functions

- def calcContoursByTriangularPotentialCuts.calcContoursByTriangluarPotentialCuts (mesh, potentialLevel↩ List, streamFunction)
- def calcContoursByTriangularPotentialCuts.getuvCutPoints (mesh, innerEdges, edgeLength, potentialCut↩ Criteria, cutPointDistanceToEdgeNode)
- def calcContoursByTriangularPotentialCuts.getRawUnsortedPoints (potentialLevelList, potentialSortedCut↩ Points)
- def calcContoursByTriangularPotentialCuts.getPotentialSortedCutPoints (potentialLevelList, uCutPoint, v↩ CutPoint)
- def calcContoursByTriangularPotentialCuts.getCutDistancesToEdgeNodes (edgeNodePotentials, potential↩ LevelList, edgeLength)
- def calcContoursByTriangularPotentialCuts.getPotentialCutCriteria (edgeNodePotentials, potentialLevelList)
- def calcContoursByTriangularPotentialCuts.getRawUnarrangedLoops (rawUnsortedPoints, innerEdges, innerEdgeOpposedNode)
- def calcContoursByTriangularPotentialCuts.checkIfPositionsElementIdenticalWithFirstList (allCurrentEdges, allCurrentOpposedNodes, checkPosition)
- def calcContoursByTriangularPotentialCuts.getEdgeOpposedNode (mesh, innerEdgesTrianglesInds, inner↩ Edges)
- def calcContoursByTriangularPotentialCuts.removeTwoElementsFromArray (array, Element1, Element2)
- def calcContoursByTriangularPotentialCuts.getInnerEdegTriangleNodes (innerEdgesTrianglesInds, mesh)
- def calcContoursByTriangularPotentialCuts.getInnerEdges (mesh)
- def calcContoursByTriangularPotentialCuts.getNumAttachedTriangles (edgeAttachedTriangles)
- def calcContoursByTriangularPotentialCuts.getEdgeAttachedTriangles (edges, mesh)
- def calcContoursByTriangularPotentialCuts.getEdges (mesh)

8.7 calcContoursByTriangularPotentialCuts.py

Go to the documentation of this file.

```
00001 import numpy as np
00002 from readMesh import updateList
00003
00004 def calcContoursByTriangluarPotentialCuts(mesh,potentialLevelList,streamFunction):
00005
00006     innerEdges, innerEdgesTrianglesInds = getInnerEdges(mesh)
00007     innerEdgeOpposedNode = getEdgeOpposedNode(mesh,innerEdgesTrianglesInds,innerEdges)
00008
00009     edgeNodePotentials = streamFunction[innerEdges]
00010     potentialCutCriteria = getPotentialCutCriteria(edgeNodePotentials,potentialLevelList)
00011     edgeLength = np.sqrt((mesh.u[innerEdges[:,0]]-mesh.u[innerEdges[:,1]])**2 +
      (mesh.v[innerEdges[:,0]]-mesh.v[innerEdges[:,1]])**2)
00012
      cutPointDistanceToEdgeNode=getCutDistancesToEdgeNodes(edgeNodePotentials,potentialLevelList,edgeLength)
00013     uCutPoint, vCutPoint =
      getuvCutPoints(mesh,innerEdges,edgeLength,potentialCutCriteria,cutPointDistanceToEdgeNode)
00014
00015     potentialSortedCutPoints = getPotentialSortedCutPoints(potentialLevelList,uCutPoint,vCutPoint)
00016     rawUnsortedPoints = getRawUnsortedPoints(potentialLevelList,potentialSortedCutPoints)
00017     rawUnarrangedLoops = getRawUnarrangedLoops(rawUnsortedPoints,innerEdges,innerEdgeOpposedNode)
00018
00019     #TODO: next step Matlab calc_contours_by trianglar_potential_cuts line 179 "evaluate for each loop
      the current orientation"
00020
00021     contours=0
00022     return contours
00023
00024 def getuvCutPoints(mesh,innerEdges,edgeLength,potentialCutCriteria,cutPointDistanceToEdgeNode):
00025     "'returns ucutPoint and vCutPoint"'
00026     uKompEdgeVec = mesh.u[innerEdges[:,1]] - mesh.u[innerEdges[:,0]]
00027     vKompEdgeVec = mesh.v[innerEdges[:,1]] - mesh.v[innerEdges[:,0]]
00028     uCutPoint,vCutPoint=[],[]
00029     for x in range(len(edgeLength)):
00030         uCutPoint.append(potentialCutCriteria[x] * (mesh.u[innerEdges[:,0]][x] +
      cutPointDistanceToEdgeNode[x]/edgeLength[x] *uKompEdgeVec[x]))
00031         vCutPoint.append(potentialCutCriteria[x] * (mesh.v[innerEdges[:,0]][x] +
      cutPointDistanceToEdgeNode[x]/edgeLength[x] *vKompEdgeVec[x]))
00032     return uCutPoint, vCutPoint
00033
```

```
00034 def getRawUnsortedPoints(potentialLevelList,potentialSortedCutPoints):
00035     "'returns rawUnsortedPoints"'
00036     rawUnsortedPoints = []
00037     for i in range(len(potentialLevelList)):
00038         rawUnsortedPointsDict = {
00039             "potential": potentialLevelList[i],
00040             "edgeInd": np.array(potentialSortedCutPoints[i])[:,2].astype(int),
00041             "uv":
      [np.array(potentialSortedCutPoints[i])[:,0],np.array(potentialSortedCutPoints[i])[:,1]],
00042         }
00043         rawUnsortedPoints.append(rawUnsortedPointsDict)
00044     return rawUnsortedPoints
00045
00046 def getPotentialSortedCutPoints(potentialLevelList,uCutPoint,vCutPoint):
00047     "'returns potentialSortedCutPoints"'
00048     potentialSortedCutPoints = []
00049     for potInd in range(len(potentialLevelList)):
00050         listelement = []
00051         for edgeInd in range(len(uCutPoint)):
00052             if uCutPoint[edgeInd][potInd] != 0:
00053
      listelement.append([uCutPoint[edgeInd][potInd],vCutPoint[edgeInd][potInd],int(edgeInd)])
00054         potentialSortedCutPoints.append(listelement)
00055     return potentialSortedCutPoints
00056
00057 def getCutDistancesToEdgeNodes(edgeNodePotentials,potentialLevelList,edgeLength):
00058     "' returns the cutPointDistanceToEdgeNode"'
00059     edgePotentialSpan = edgeNodePotentials[:,1]-edgeNodePotentials[:,0]
00060     cutPointDistanceToEdgeNode=[]
00061     for x in range(len(edgeNodePotentials)):
00062         cutPointDistanceToEdgeNodePart = []
00063         for y in range(len(potentialLevelList)):
00064             cutPointDistanceToEdgeNodePart.append(np.abs(edgeLength[x]/edgePotentialSpan[x] *
      (potentialLevelList[y]-edgeNodePotentials[x][0])))
00065         cutPointDistanceToEdgeNode.append(cutPointDistanceToEdgeNodePart)
00066     return cutPointDistanceToEdgeNode
00067
00068
00069 def getPotentialCutCriteria(edgeNodePotentials,potentialLevelList):
00070     "'returns the PotentialCutCriteria"'
00071     minEdgePotential=[]
00072     maxEgdePotential = []
00073     for x in range(len(edgeNodePotentials)):
00074         minEdgePotential.append(min(edgeNodePotentials[x]))
00075         maxEgdePotential.append(max(edgeNodePotentials[x]))
00076     triBelowPotStep = []
00077     triAbovePotStep = []
00078     for i in range(len(maxEgdePotential)):
00079         triBelowPotStep.append(maxEgdePotential[i]>potentialLevelList)
00080         triAbovePotStep.append(minEdgePotential[i]<potentialLevelList)
00081     return np.array(triBelowPotStep) & np.array(triAbovePotStep)
00082
00083 def getRawUnarrangedLoops(rawUnsortedPoints,innerEdges,innerEdgeOpposedNode):
00084     "'returns rawUnarrangedLoops"'
00085     rawUnarrangedLoopsTotal = []
00086     for potentialGroupInd in range(len(rawUnsortedPoints)):
00087             allCurrentEdges = innerEdges[rawUnsortedPoints[potentialGroupInd]['edgeInd']]
00088             allCurrentOpposedNodes =
      np.array(innerEdgeOpposedNode)[rawUnsortedPoints[potentialGroupInd]['edgeInd']]
00089             allCurrentUVKoords = rawUnsortedPoints[potentialGroupInd]['uv']
00090             setNewStart = True
00091             numBuildLoops = 0
00092             edgeAlreadyUsed = np.zeros(len(allCurrentEdges))
00093             rawUnarrangedLoops = []
00094
00095             while not edgeAlreadyUsed.all():
00096                 if setNewStart:
00097                     oneLoop =[]
00098                     numBuildLoops += 1
00099                     startingEdge = min(np.argwhere(edgeAlreadyUsed== 0))
00100                     rawUnarrangedLoopsDict = {
00101                         "edgeInd":allCurrentEdges[startingEdge],
00102                         "uv": np.array(allCurrentUVKoords)[:,startingEdge]
00103                     }
00104                     oneLoop.append(rawUnarrangedLoopsDict)
00105                     edgeAlreadyUsed[startingEdge] = 1
00106                     currentEdge = startingEdge
00107
00108
      testElement,testtestElement=checkIfPositionsElementIdenticalWithFirstList(allCurrentEdges,allCurrentOpposedNodes,
      currentEdge)
00109                     neighbouringFreeNextEdges = np.argwhere(np.any(testElement, axis=1) &
      np.any(testtestElement, axis=1))
00110                     if not neighbouringFreeNextEdges.any(): break
00111                     elif len(neighbouringFreeNextEdges) == 1:
00112                         setNewStart = False
00113                         nextEdge = neighbouringFreeNextEdges[0]
```

```
00114                     else:
00115                         setNewStart = False
00116                         if not edgeAlreadyUsed[neighbouringFreeNextEdges[0]]: nextEdge =
      neighbouringFreeNextEdges[0]
00117                         else: nextEdge = neighbouringFreeNextEdges[1]
00118
00119                 while not (nextEdge == startingEdge):
00120                     rawUnarrangedLoopsDict = {
00121                         "edgeInd":allCurrentEdges[nextEdge],
00122                         "uv": np.array(allCurrentUVKoords)[:,nextEdge]
00123                     }
00124                     oneLoop.append(rawUnarrangedLoopsDict)
00125                     edgeAlreadyUsed[nextEdge] = 1
00126                     currentEdge = np.copy(nextEdge)
00127
00128
      testElement,testtestElement=checkIfPositionsElementIdenticalWithFirstList(allCurrentEdges,allCurrentOpposedNodes,
      currentEdge)
00129                     possibleNextEdges = np.argwhere(np.any(testElement, axis=1) &
      np.any(testtestElement, axis=1))
00130                     possibleNextEdges = np.setdiff1d(possibleNextEdges,np.argwhere(edgeAlreadyUsed ==
      1))
00131                     if not possibleNextEdges.any():
00132                         rawUnarrangedLoops.append(oneLoop)
00133                         break
00134                     elif len(possibleNextEdges) == 1: nextEdge = possibleNextEdges[0]
00135                     else:
00136                         if not edgeAlreadyUsed[possibleNextEdges[0]]: nextEdge = possibleNextEdges[0]
00137                         else: nextEdge = possibleNextEdges[1]
00138                 setNewStart = True
00139             rawUnarrangedLoopsTotal.append(rawUnarrangedLoops)
00140     return rawUnarrangedLoopsTotal
00141
00142 def checkIfPositionsElementIdenticalWithFirstList(allCurrentEdges, allCurrentOpposedNodes,
      checkPosition):
00143     "'returns lists of elements with booleans if checkPostions Element of first and secound Element
      are identical with allCurrentEdges"'
00144     testElement=[]
00145     testtestElement=[]
00146     for i in allCurrentEdges:
00147         testElement1 =[]
00148         testElement2 =[]
00149         for j in i:
00150             testElement1.append(np.any(j == allCurrentOpposedNodes[checkPosition]))
00151             testElement2.append(np.any(j == allCurrentEdges[checkPosition]))
00152         testElement.append(testElement1)
00153         testtestElement.append(testElement2)
00154     return testElement,testtestElement
00155
00156 def getEdgeOpposedNode(mesh,innerEdgesTrianglesInds,innerEdges):
00157     "'returns the to the edge opposed node for each attached triangle"'
00158     innerEdgeTriangleNodes = getInnerEdegTriangleNodes(innerEdgesTrianglesInds,mesh)
00159     edgeOpposedNodes = []
00160     for edgeInd in range(len(innerEdges)):
00161         oneEdgeOpposedNodes= []
00162         for triangleInd in range(len(innerEdgeTriangleNodes[edgeInd])):
00163
      oneEdgeOpposedNodes.append(removeTwoElementsFromArray(innerEdgeTriangleNodes[edgeInd][triangleInd],innerEdges[edgeInd]
00164         edgeOpposedNodes.append(oneEdgeOpposedNodes)
00165     return edgeOpposedNodes
00166
00167 def removeTwoElementsFromArray(array,Element1,Element2):
00168     "'returns the array without the two elements."'
00169     return updateList(updateList(array,Element1),Element2)
00170
00171 def getInnerEdegTriangleNodes(innerEdgesTrianglesInds,mesh):
00172     "'returns the Node Indices for the Triangles."'
00173     innerEdgeTriangleNodes = []
00174     for edgeInd in range(len(innerEdgesTrianglesInds)):
00175
      innerEdgeTriangleNodes.append([np.array(mesh.faces[innerEdgesTrianglesInds[edgeInd][0]]),np.array(mesh.faces[innerEdges
00176     return innerEdgeTriangleNodes
00177
00178 def getInnerEdges(mesh):
00179     "'returns the inner edges an the corresponding triangleInds."'
00180     edges = getEdges(mesh)
00181     edgeAttachedTriangles = getEdgeAttachedTriangles(edges,mesh)
00182     numAttachedTriangles = getNumAttachedTriangles(edgeAttachedTriangles)
00183     innerEdges =[]
00184     innerEdgesTrianglesInds = []
00185     for index in range(len(edges)):
00186         if numAttachedTriangles[index] == 2:
00187             innerEdges.append(edges[index])
00188             innerEdgesTrianglesInds.append(edgeAttachedTriangles[index])
00189     return np.array(innerEdges), innerEdgesTrianglesInds
00190
00191 def getNumAttachedTriangles(edgeAttachedTriangles):
```

```
00192     "'returns the number of attached triangles per edge."'
00193     numAttachedTriangles =[]
00194     for edgeInd in range(len(edgeAttachedTriangles)):
00195         numAttachedTriangles.append(len(edgeAttachedTriangles[edgeInd]))
00196     return numAttachedTriangles
00197
00198 def getEdgeAttachedTriangles(edges,mesh):
00199     "'returns the attached triangles for each edge."'
00200     allEdgesTriangles=[]
00201     for edge in edges:
00202         edgeFaces =[]
00203         for triangleIndex in range(len(mesh.faces)):
00204             if edge[0] in mesh.faces[triangleIndex] and edge[1] in mesh.faces[triangleIndex]:
00205                 edgeFaces.append(triangleIndex)
00206         allEdgesTriangles.append(edgeFaces)
00207     return allEdgesTriangles
00208
00209 def getEdges(mesh):
00210     "'returns all edges in the mesh."'
00211     edges=[]
00212     for node in range(len(mesh.vertices)):
00213         for triangle in mesh.faces:
00214             if node in triangle:
00215                 if node == triangle[0]:
00216                     if node < triangle[1] and [triangle[0],triangle[1]] not in edges:
00217                         edges.append([triangle[0],triangle[1]])
00218                     if node < triangle[2] and [triangle[0],triangle[2]] not in edges:
00219                         edges.append([triangle[0],triangle[2]])
00220                 elif node == triangle[1]:
00221                     if node < triangle[0] and [triangle[1],triangle[0]] not in edges:
00222                         edges.append([triangle[1],triangle[0]])
00223                     if node < triangle[2] and [triangle[1],triangle[2]] not in edges:
00224                         edges.append([triangle[1],triangle[2]])
00225                 elif node == triangle[2]:
00226                     if node < triangle[0] and [triangle[2],triangle[0]] not in edges:
00227                         edges.append([triangle[2],triangle[0]])
00228                     if node < triangle[1] and [triangle[2],triangle[1]] not in edges:
00229                         edges.append([triangle[2],triangle[1]])
00230
00231     return edges
```

8.8 subfunctions/calcPotentialLevels.py File Reference

Namespaces

- namespace calcPotentialLevels

Functions

- def calcPotentialLevels.calcPotentialLevels (streamFunction, numLevels, levelOffset)

8.9 calcPotentialLevels.py

Go to the documentation of this file.

```
00001 import numpy as np
00002
00003 def calcPotentialLevels(streamFunction, numLevels, levelOffset):
00004     "'returns the contourStep and the potentialLevelList. more options need to be implemented, if
      there is more than one mesh."'
00005     SFRangePerMesh = np.max(streamFunction)-np.min(streamFunction)
00006     contourStep = SFRangePerMesh/(numLevels-1+2*levelOffset)
00007     potentialLevelList=
      np.arange(0,numLevels)*contourStep+(np.min(streamFunction)+levelOffset*contourStep)
00008
00009     return contourStep, potentialLevelList
```

8.10 subfunctions/defineTargetField.py File Reference

Classes

- class defineTargetField.TargetField
- class defineTargetField.TargetFieldGiven

Namespaces

- namespace defineTargetField

Functions

- def defineTargetField.distanceBetweenPoints (point1, point2)

8.11 defineTargetField.py

Go to the documentation of this file.

```
00001 #kugel in zylinder, gleichmäßig im Volumen verteilte Punkte, mit gewünschtem Feld
00002 import numpy as np
00003 import pandas as pd
00004 import matplotlib.pyplot as plt
00005 from readMesh import getMeshFromSTL
00006
00007
00008 class TargetField():
00009     def __init__(self,center,radius,direction):#direction: 0==x,1==y,2==z
00010         self.center = center
00011         self.radius = radius
00012         self.vertices = self.getTargetPoints()
00013         self.fieldValues = self.getMagneticFieldValues(direction)
00014
00015     def getTargetPoints(self):
00016         "'generate Target Points within a circle with the given specifications"'
00017         newX = np.linspace(self.center[0]-self.radius,self.center[0]+self.radius,25)
00018         newY = np.linspace(self.center[1]-self.radius,self.center[1]+self.radius,25)
00019         newZ = np.linspace(self.center[2]-self.radius,self.center[2]+self.radius,25)
00020         newX,newY,newZ = np.meshgrid(newX, newY, newZ, indexing='ij')
00021         targetVertices = []
00022         for i in range(len(newX)):
00023             for j in range(len(newX)):
00024                 for k in range(len(newX[i][j])):
00025                     if distanceBetweenPoints(self.center,[newX[i][j][k],newY[i][j][k],newZ[i][j][k]])
    <= self.radius:
00026                         targetVertices.append([newX[i][j][k],newY[i][j][k],newZ[i][j][k]])
00027                     else: continue
00028         return np.array(targetVertices)#array is important to be able to do [:,0]
00029
00030     def getMagneticFieldValues(self,direction):
00031         "'returns magnetic field values analogus to matlab skript"'
00032         targetStrength = 0.1
00033         targetField = np.zeros((3,len(self.vertices)))
00034         targetField[2] = self.vertices[:,direction]
00035         targetField = targetField*targetStrength
00036         return targetField
00037
00038 class TargetFieldGiven(TargetField):
00039     def __init__(self,filename,direction):
00040         self.vertices,self.faces = getMeshFromSTL(filename)
00041         self.fieldValuesfieldValues = self.getMagneticFieldValues(direction)
00042
00043 def distanceBetweenPoints(point1,point2):
00044     "'returns the distance between two given points"'
00045     result = 0
00046     for i in range(len(point1)):
00047         result += (point1[i]-point2[i])**2
00048     return np.sqrt(result)
00049
00050
00056
00057 #ax.scatter3D(points[0],points[1],points[2])
00058 #plt.show()
```

8.12 subfunctions/readMesh.py File Reference

Classes

- class readMesh.CylindricMesh
- class readMesh.CylindricMeshGiven

Namespaces

- namespace readMesh

Functions

- def readMesh.calculateNormal (vec)
- def readMesh.updateList (edgeList, otheredge)
- def readMesh.getMeshFromSTL (filename)
- def readMesh.checkIfVecInVeclist (node, vecList)

8.13 readMesh.py

Go to the documentation of this file.

```
00001 import numpy as np
00002 import matplotlib.pyplot as plt
00003 import meshzoo
00004 import trimesh
00005 import scipy
00006
00007 def calculateNormal(vec):
00008     "'returns the norm for a given 3d vector"'
00009     if len(vec) == 3:
00010         v1 = vec[1] - vec[0]
00011         v2 = vec[2] - vec[0]
00012         return
       np.cross(v1,v2)/np.sqrt(np.cross(v1,v2)[0]**2+np.cross(v1,v2)[1]**2+np.cross(v1,v2)[2]**2)
00013     else:
00014         print("Mesh-Generation is going wrong. Faces do not have 3 components")
00015         return False
00016
00017 def updateList(edgeList,otheredge):
00018     "'returns the edgeList without otheredge"'
00019     return [a for a, skip in zip(edgeList, [np.allclose(a, otheredge) for a in edgeList]) if not skip]
00020
00021 #option 1: create mesh
00022 class CylindricMesh():
00023     def __init__(self,coilLength,coilRadius,n):
00024         self.vertices, self.faces = meshzoo.tube(length=coilLength, radius=coilRadius,
       n=int(n))#points, cells(index of the points that close the cell)
00025         self.vertices = np.array(self.vertices)
00026         self.normals=self.getNormals()
00027         self.openBoundaries=self.getOpenBoundaries()
00028         self.areas = self.getAreas()
00029         self.current = self.getCurrent()
00030         self.neighbours=self.getNeighbourTriangleIndices()
00031         self.rotatedCaylinder=self.getRotatedCopy()
00032         self.u,self.v=self.get2Dcoordinates()
00033         self.neighbourareas = self.getNeighbourAreas()
00034         self.currentDensityFaces=[]
00035         self.vertexNormals=self.getVertexNormals()
00036         self.boundary = self.checkIfBoundary()
00037         self.oneRingList = self.getOneRingList()
00038         self.neighbourcurrents = self.getNeighbourCurrents()
00039         self.neighbourcurrentUnsorted = self.getNeighbourCurrentsUnsorted()
00040
00041     def checkIfBoundary(self):
00042         "'returns a list of boolean if the vertice is a boundary vertice"'
00043         boundaryBooleans = []
00044         for nodeElements in range(len(self.vertices)):
```

```
00045             boundaryBooleans.append((nodeElements in self.openBoundaries[0]) | (nodeElements in
      self.openBoundaries[1]))
00046         return boundaryBooleans
00047
00048     def getVertexNormals(self):
00049         "'returns the normals of the vertices. These are calculated as average of the touching faces
      normals."'
00050         vertexNormals = []
00051         for vertex in range(len(self.vertices)):
00052             sum = 0
00053             for x in np.array(self.neighbours[vertex]):
00054                 sum+= self.normals[x]
00055             vertexNormals.append(sum/len(self.neighbours[vertex]))
00056         return vertexNormals
00057
00058     def getNormals(self):
00059         "'returns the normals of the faces"'
00060         normals=[]
00061         for i in range(len(self.faces)):
00062             normals.append(calculateNormal(self.vertices[self.faces[i]]))
00063         return normals
00064
00065     def getNeighbourAreas(self):
00066         "'returns the areas of the neighbour triangles for every node"'
00067         neighbourareas = []
00068         for i in range(len(self.vertices)):
00069             neighbourareasparts=[]
00070             for j in self.neighbours[i]:
00071                 neighbourareasparts.append(self.areas[j])
00072             neighbourareas.append(neighbourareasparts)
00073         return neighbourareas
00074
00075     def getNeighbourCurrents(self):
00076         "'returns the currents of the neighbour triangles for every node"'
00077         neighbourcurrents = []
00078         for i in range(len(self.vertices)):
00079             neighbourcurrentparts=[]
00080             for j in self.oneRingList[i]:
00081                 neighbourcurrentparts.append((self.vertices[j[1]] -
      self.vertices[j[0]])/(scipy.linalg.norm(np.cross(self.vertices[j[1]]-self.vertices[i],self.vertices[j[0]]
      - self.vertices[i]))))
00082             neighbourcurrents.append(neighbourcurrentparts)
00083         return neighbourcurrents
00084
00085     def getNeighbourCurrentsUnsorted(self):
00086         "'returns the currents of the neighbour triangles for every node before sorting"'
00087         neighbourcurrents = []
00088         for i in range(len(self.vertices)):
00089             neighbourcurrentparts=[]
00090             for j in self.neighbours[i]:
00091                 neighbourcurrentparts.append(self.current[j])
00092             neighbourcurrents.append(neighbourcurrentparts)
00093         return neighbourcurrents
00094
00095     def getCurrent(self):
00096         "'returns the current for the triangles made with the points in faces
00097         C = (c-b)/(2*Fläche)"'
00098         current =[]
00099         for i in range(len(self.areas)):
00100
      current.append((self.vertices[self.faces[i][2]]-self.vertices[self.faces[i][1]])/(2*self.areas[i]))
00101         return current
00102
00103     def getAreas(self):
00104         "'returns the areas of the triangles made with the points in faces"'
00105         areas=[]
00106         for i in self.faces:
00107
      areas.append(np.linalg.norm(np.cross((self.vertices[i[1]]-self.vertices[i[0]]),(self.vertices[i[2]]-self.vertices[i[0]
00108         return areas
00109
00110     def getOpenBoundaries(self):
00111         "'returns indexes of the nodes at the edges of a cylinder extended in z-direction"'
00112         max = 0
00113         upperopen=[]
00114         min = 0
00115         loweropen=[]
00116         for i in range(len(self.vertices)):
00117             if self.vertices[i][2]< min:
00118                 min = self.vertices[i][2]
00119                 loweropen = [i]
00120             elif self.vertices[i][2] == min:
00121                 loweropen.append(i)
00122             elif self.vertices[i][2]>max:
00123                 max = self.vertices[i][2]
00124                 upperopen = [i]
00125             elif self.vertices[i][2]==max:
```

```
00126                 upperopen.append(i)
00127             else: continue
00128         return [upperopen,loweropen]#TODO tauschen
00129
00130     def getBoundaryEdges(self):
00131         "'returns the nodes for each boundary in the correct order"'
00132         boundaryEdges=[]
00133         for boundary in self.openBoundaries:
00134             eachBoundaryEdges=[]
00135             for boundaryPoint in boundary:
00136                 for neighbour in self.faces[self.neighbours[boundaryPoint]]:
00137                     for neighbournode in neighbour:
00138                         if neighbournode in boundary and neighbournode != boundaryPoint:
00139                             eachBoundaryEdges.append([boundaryPoint,neighbournode])
00140             eachBoundaryEdges = self.removeDoubleEdges(eachBoundaryEdges)
00141             boundaryEdges.append(eachBoundaryEdges)
00142         return boundaryEdges
00143
00144     def removeDoubleEdges(self,edgeList):
00145         "'returns the edgeList with each edge just once."'
00146         newList=np.copy(edgeList)
00147         for edgeInd in range(len(newList)):
00148             for otheredgeInd in range(len(newList)):
00149                 if (newList[edgeInd][::-1] == newList[otheredgeInd]).all() and edgeInd < otheredgeInd:
00150                     edgeList = updateList(edgeList,newList[otheredgeInd])
00151         return edgeList
00152
00153     def getBoundaryLoopNodes(self):
00154         "'returns the unsorted nodes for the boundaryLoop."'
00155         boundaryEdges = self.getBoundaryEdges()
00156         boundaryEdges = self.turnAnsSortElements(boundaryEdges)
00157         boundaryEdges = [np.flip(boundaryEdges[1]),boundaryEdges[0]] #TODO welche Regelmäßigkeit,
    warum so?
00158         boundaryLoopNodes=[]
00159         for boundary in boundaryEdges:
00160             boundaryNodes = np.append(np.array(boundary)[:,0],boundary[0][0])
00161             boundaryLoopNodes.append(boundaryNodes)
00162         return boundaryLoopNodes
00163
00164     def getRotatedCopy(self):
00165         "'returns rotated copy of the vertices. If the cylinder is orientated along the z axis we need
    a rotated copy."'
00166         boundaryLoopNodes = self.getBoundaryLoopNodes()
00167         rotationVec,angle = self.calcRotationVec(boundaryLoopNodes)
00168         rotMat = self.calc3DRotMatByVec(rotationVec,angle)
00169         rotatedVertices = self.getRotatedVertices(rotMat)
00170         return rotatedVertices
00171
00172     def getRotatedVertices(self,rotMat):
00173         "'returns the rotated vertices (multiplication with rotMat)."'
00174         rotatedVertices = []
00175         for i in self.vertices:
00176             rotatedVertices.append(np.dot(rotMat,np.transpose(i)))
00177         return rotatedVertices
00178
00179     def calcRotationVec(self,boundaryLoopNodes):
00180         "'returns the rotationVector and the angle based on the boundaryLoopNodes."'
00181         openingMean =
    [np.mean(self.vertices[boundaryLoopNodes[0]][:,0]),np.mean(self.vertices[boundaryLoopNodes[0]][:,1]),np.mean(self.vert
00182         overallMean = np.mean(self.vertices)
00183         oldOrientationVec=(openingMean-overallMean)/np.linalg.norm(openingMean-overallMean)
00184         zVec = [0,0,1]
00185         sina =
    np.linalg.norm(np.cross(oldOrientationVec,zVec))/(np.linalg.norm(oldOrientationVec)*np.linalg.norm(zVec))
00186         cosa =
    np.linalg.norm(np.dot(oldOrientationVec,zVec))/(np.linalg.norm(oldOrientationVec)*np.linalg.norm(zVec))
00187         angle = np.arctan2(sina,cosa)
00188         rotationVec =
    np.cross(oldOrientationVec,zVec)/np.linalg.norm(np.cross(oldOrientationVec,zVec))
00189         return rotationVec,angle
00190
00191     def calc3DRotMatByVec(self,rotationVec,angle):
00192         "'returns the rotation matrix calculated from the rotationVector"'
00193         uX,uY,uZ = rotationVec
00194         tmp1 = np.sin(angle)
00195         tmp2 = np.cos(angle)
00196         tmp3 = (1-np.cos(angle))
00197         rotMat = np.zeros((3,3))
00198         rotMat[0][0] = tmp2 + uX*uX*tmp3
00199         rotMat[0][1] = uX*uY*tmp3-uZ*tmp1
00200         rotMat[0][2] = uX*uZ*tmp3+uY*tmp1
00201         rotMat[1][0] = uY*uX*tmp3+uZ*tmp1
00202         rotMat[1][1] = tmp2+uY*uY*tmp3
00203         rotMat[1][2] = uY*uZ*tmp3-uX*tmp1
00204         rotMat[2][0] = uZ*uX*tmp3-uY*tmp1
00205         rotMat[2][1] = uZ*uX*tmp3+uX*tmp1
00206         rotMat[2][2] = tmp2+uZ*uZ*tmp3
```

```
00207         return rotMat
00208
00209     def turnAnsSortElements(self,boundaryEdges):
00210         "'returns the given list in sorted. If needed single elements were turned to close the loop."'
00211         new=[]
00212         for boundary in boundaryEdges:
00213             start = boundary[0]
00214             newElement = [start]
00215             while len(newElement) < len(boundary):
00216                 for element in boundary:
00217                     if start == element:
00218                         continue
00219                     elif start[1] == element[0]:
00220                         if element[1] is not start[0]:
00221                             newElement.append(element)
00222                             break
00223                     elif start[1] == element[1]:
00224                         if start[0] is not element[1]:
00225                             newElement.append([element[1],element[0]])
00226                             break
00227                 start = newElement[-1]
00228             new.append(newElement)
00229         return new
00230
00231     def get2Dcoordinates(self):
00232         "'returns the from 3D to 2D converted vertices"'
00233         corods = np.array(self.rotatedCaylinder)
00234         minZCylinder = min(corods[:,2])
00235         corods[:,2] = corods[:,2]+minZCylinder
00236         phiCoord = np.arctan2(corods[:,1],corods[:,0])
00237         r = np.sqrt(corods[:,0]**2+corods[:,1]**2)
00238         u = (corods[:,2]-np.mean(r))*np.sin(phiCoord)
00239         v = (corods[:,2]- np.mean(r))*np.cos(phiCoord)
00240         return u,v
00241
00242     def getNeighbourTriangleIndices(self):
00243         "'returns the indices of the neighbour triangles of every node"'
00244         neighbourtrianglesIndices=[]
00245         for node in self.vertices:
00246             k=[]
00247             for i in range(len(self.faces)):
00248                 vecList = self.vertices[self.faces[i]]
00249                 if checkIfVecInVeclist(node,vecList):
00250                     k.append(i)
00251             neighbourtrianglesIndices.append(k)
00252         return neighbourtrianglesIndices
00253
00254     def getOneRingList(self):
00255         "'returns sorted list with nodes around every node"'
00256         oneRingList = self.createOneRingList()
00257         oneRingList = self.ensureUniformOrientation(oneRingList)
00258         oneRingList = self.orderElementsInCircularArangement(oneRingList)
00259         return oneRingList
00260
00261     def orderElementsInCircularArangement(self,oneRingList):
00262         "'returns the List in a circular arrangement"'
00263         for nodeElements in range(len(oneRingList)):
00264             if self.boundary[nodeElements]:
00265                 start = self.findStartInBoundaryCase(oneRingList,nodeElements)
00266             else:
00267                 start = oneRingList[nodeElements][0]
00268
00269             new = self.arrangeCircular(start,oneRingList[nodeElements])
00270             oneRingList[nodeElements] = new
00271         return oneRingList
00272
00273     def arrangeCircular(self,start,Elements):
00274         "'returns the elements in a circular order beginning with start"'
00275         new = [start]
00276         while len(new) != len(Elements):
00277             for i in Elements:
00278                 if len(new) == len(Elements):
00279                     break
00280                 elif new[-1][1] == i[0]:
00281                     new.append(i)
00282         return new
00283
00284     def findStartInBoundaryCase(self,oneRingList,nodeNumber):
00285         "'returns the correct start triangle for ordering the triangles around a boundary vertice"'
00286         index = 0
00287         start = oneRingList[nodeNumber][0]
00288         correctstart = self.checkStartTriangle(oneRingList[nodeNumber],start)
00289         while not correctstart:
00290             start = oneRingList[nodeNumber][index+1]
00291             correctstart = self.checkStartTriangle(oneRingList[nodeNumber],start)
00292             index+=1
00293         return start
```

```
00294
00295     def checkStartTriangle(self,verticeTriangles,start):
00296         "'returns boolean if "start" is the correct startTriangle"'
00297         test = []
00298         for i in range(len(verticeTriangles)):
00299             test.append(start[0]==verticeTriangles[i][1])
00300         return not np.any(test)
00301
00302     def createOneRingList(self):
00303         "'returns a list with the other two triangle Points for each triangle per node"'
00304         indices = self.getNeighbourTriangleIndices()
00305         oneRingList=[]
00306         for i in range(len(indices)):
00307             eachnode=[]
00308             for k in range(len(indices[i])):
00309                 new=[]
00310                 for j in range(3):
00311                     if self.faces[indices[i][k]][j] != i:
00312                         new.append(self.faces[indices[i][k]][j])
00313                 eachnode.append(new)
00314             oneRingList.append(eachnode)
00315         return oneRingList
00316
00317     def ensureUniformOrientation(self,oneRingList):
00318         "'returns oneRingList with ensured uniform Orientation"'
00319         for nodeelements in range(len(oneRingList)):
00320             for neighbournodes in oneRingList[nodeelements]:
00321                 b = self.vertices[neighbournodes[0]]
00322                 c = self.vertices[neighbournodes[1]]
00323                 a = self.vertices[nodeelements]
00324                 crossVec = np.cross(c-b,b-a)
00325
00326                 if np.sign(np.dot(self.vertexNormals[nodeelements],crossVec)) > 0:
00327                     before0 = neighbournodes[0]
00328                     before1 = neighbournodes[1]
00329                     neighbournodes[0] = before1
00330                     neighbournodes[1] = before0
00331         return oneRingList
00332
00333
00334 #option 2: create cylindric mesh
00335 def getMeshFromSTL(filename):
00336     "'returns vertices and faces from given stl file meshes"'
00337     myobj = trimesh.load_mesh(filename, enable_post_processing=True, solid=True)
00338     return myobj.vertices, myobj.faces
00339
00340 class CylindricMeshGiven(CylindricMesh):
00341     def __init__(self,filename):
00342         self.verticesvertices,self.facesfaces = getMeshFromSTL(filename)
00343         self.verticesvertices = np.array(self.verticesvertices)
00344         self.normalsnormals=self.getNormals()
00345         self.openBoundariesopenBoundaries=self.getOpenBoundaries()
00346         self.areasareas = self.getAreas()
00347         self.currentcurrent = self.getCurrent()
00348         self.neighboursneighbours=self.getNeighbourTriangleIndices()
00349         self.rotatedCaylinderrotatedCaylinder=self.getRotatedCopy()
00350         self.u,self.vv=self.get2Dcoordinates()
00351         self.neighbourareasneighbourareas = self.getNeighbourAreas()
00352         self.currentDensityFacescurrentDensityFaces=[]
00353         self.vertexNormalsvertexNormals=self.getVertexNormals()
00354         self.boundaryboundary = self.checkIfBoundary()
00355         self.oneRingListoneRingList = self.getOneRingList()
00356         self.neighbourcurrentsneighbourcurrents = self.getNeighbourCurrents()
00357         self.neighbourcurrentUnsortedneighbourcurrentUnsorted = self.getNeighbourCurrentsUnsorted()
00358
00359 def checkIfVecInVeclist(node,vecList):
00360     "'returns Boolean if a 3 components vec is in a list of 3 component elements"'
00361     return (node == vecList[0]).all()|(node == vecList[1]).all()|( node == vecList[2]).all()
00362
00363
00364
00365
00366
00369
00370 #x,y = mesh.get2Dcoordinates()
00371 #plt.plot(x,y,'.')
00372 #plt.show()
00373
00374
00375
00378
00379 # X=[]
00380 # Y=[]
00381 # Z=[]
00382 # print(np.shape(mesh.vertices))
00383 # for i in range(len(mesh.vertices)):
00384 #     X.append(mesh.vertices[i][0])
```

```
00385 #     Y.append(mesh.vertices[i][1])
00386 #     Z.append(mesh.vertices[i][2])
00387 # ax.scatter3D(X,Y,Z)
00388 # plt.show()
00389
```

8.14 subfunctions/resistanceMatrix.py File Reference

Namespaces

- namespace resistanceMatrix

Functions

- def resistanceMatrix.getResistanceMatrix (test, mesh, materialFactor)
- def resistanceMatrix.formFinalResistanceMat (resistanceMatrix, materialFactor)
- def resistanceMatrix.createPreviousResistanceMat (mesh, matElementsShouldGetValue)
- def resistanceMatrix.getResistanceSumForDifferent (mesh, trianglesWithBothNodes, nodeInd1, nodeInd2)
- def resistanceMatrix.getResistanceSumForSame (mesh, nodeInd1)
- def resistanceMatrix.calculateArea (Point1, Point2, Point3)
- def resistanceMatrix.calculateCurrent (Point1, Point2, Point3)
- def resistanceMatrix.getMatElementShouldGetValue (mesh)
- def resistanceMatrix.getPartnerElement (triangle, trianglesWithBothNodes, nodeInd2)
- def resistanceMatrix.elementInArray (array, value)
- def resistanceMatrix.elementInAandB (a, b)
- def resistanceMatrix.compareMultipleElementsBoolean (elements, testelements)
- def resistanceMatrix.getSpatialDistancesMatrix (mesh)
- def resistanceMatrix.getNeighbourhoodMatrix (mesh)
- def resistanceMatrix.compareMultipleElementsBooleanTest ()

8.15 resistanceMatrix.py

Go to the documentation of this file.

```
00001 import numpy as np
00002 from defineTargetField import distanceBetweenPoints
00003
00004 def getResistanceMatrix(test,mesh,materialFactor):
00005     "'returns the resistance Matrix for the given mesh"'
00006     matElementsShouldGetValue = getMatElementShouldGetValue(mesh)
00007     test.matElementsShouldGetValue = matElementsShouldGetValue
00008     resistanceMatrix = createPreviousResistanceMat(mesh,matElementsShouldGetValue)
00009     resistanceMatrix = formFinalResistanceMat(resistanceMatrix,materialFactor)
00010     return resistanceMatrix
00011
00012 def formFinalResistanceMat(resistanceMatrix,materialFactor):
00013     "'returns resistanceMatrix in final form. added with its transposed and multiplied with
      materialFactor"'
00014     resistanceMatrix = (resistanceMatrix + np.transpose(resistanceMatrix))*materialFactor
00015     return resistanceMatrix
00016
00017 def createPreviousResistanceMat(mesh,matElementsShouldGetValue):
00018     "'returns resistanceMatrix mit entries in matElementsShouldGetValue"'
00019     resistanceMatrix = np.zeros((len(mesh.vertices),len(mesh.vertices)),dtype=float)
00020     for elementIndex in range(len(matElementsShouldGetValue[0])):
00021         nodeInd1 = int(matElementsShouldGetValue[0][elementIndex])
00022         nodeInd2 = int(matElementsShouldGetValue[1][elementIndex])
00023         if nodeInd1 == nodeInd2:
00024             resistanceSum = getResistanceSumForSame(mesh,nodeInd1)
00025             resistanceMatrix[nodeInd1][nodeInd1] = resistanceSum
00026         else:
```

```
00027             trianglesWithBothNodes =
      [elementInArray(mesh.oneRingList[nodeInd1],nodeInd2),elementInArray(mesh.oneRingList[nodeInd2],nodeInd1)]
00028             resistanceSum =
      getResistanceSumForDifferent(mesh,trianglesWithBothNodes,nodeInd1,nodeInd2)
00029             resistanceMatrix[nodeInd1][nodeInd2] = resistanceSum
00030     return resistanceMatrix
00031
00032 def getResistanceSumForDifferent(mesh,trianglesWithBothNodes,nodeInd1,nodeInd2):
00033     "'returns resistanceSum for the Case node1 and node2 are different"'
00034     resistanceSum =0
00035     for triangle in trianglesWithBothNodes[0]:
00036         triangelArea =
      calculateArea(mesh.vertices[nodeInd1],mesh.vertices[triangle[0]],mesh.vertices[triangle[1]])
00037         primaryCurrent =
      calculateCurrent(mesh.vertices[nodeInd1],mesh.vertices[triangle[0]],mesh.vertices[triangle[1]])
00038         partnerElement = getPartnerElement(triangle,trianglesWithBothNodes,nodeInd2)
00039         secondaryCurrent =
      calculateCurrent(mesh.vertices[nodeInd2],mesh.vertices[partnerElement[0]],mesh.vertices[partnerElement[1]])
00040         resistanceSum = resistanceSum + np.dot(primaryCurrent,secondaryCurrent)*(triangelArea**2)
00041     return resistanceSum
00042
00043 def getResistanceSumForSame(mesh,nodeInd1):
00044     "'returns resistanceSum for the Case node1 and node2 are the same"'
00045     resistanceSum = 0
00046     for triangle in mesh.oneRingList[nodeInd1]:
00047         triangelArea =
      calculateArea(mesh.vertices[nodeInd1],mesh.vertices[triangle[0]],mesh.vertices[triangle[1]])
00048         current =
      calculateCurrent(mesh.vertices[nodeInd1],mesh.vertices[triangle[0]],mesh.vertices[triangle[1]])
00049         resistanceSum = resistanceSum + np.dot(current,current)*triangelArea**2
00050     return resistanceSum
00051
00052 def calculateArea(Point1,Point2,Point3):
00053     "'returns the area of a triangle with 3 given Points"'
00054     return np.linalg.norm(np.cross((Point2-Point1),(Point3-Point1)))/2
00055
00056 def calculateCurrent(Point1, Point2, Point3):
00057     "'returns the current of a triangle with 3 given Points"'
00058     return (Point3-Point2)/(np.linalg.norm(np.cross((Point2-Point1),(Point3-Point1))))
00059
00060 def getMatElementShouldGetValue(mesh):
00061     "'returns a list with positions in the matix that should get a value (diagonal elements == same
      nodes and neighbouring nodes)"'
00062     nodeAdjacencyMatrix = getNeighbourhoodMatrix(mesh)
00063     neighbourPairs = np.array(np.where(nodeAdjacencyMatrix))
00064     neighbourPairs = neighbourPairs[:,neighbourPairs[1].argsort()]
00065     matElementsShouldGetValue = [np.concatenate([np.linspace(0,263,264),neighbourPairs[0]])
      ,np.concatenate([np.linspace(0,263,264),neighbourPairs[1]])]
00066     return matElementsShouldGetValue
00067
00068 def getPartnerElement(triangle,trianglesWithBothNodes,nodeInd2):
00069     "'returns the specified triangle both nodes are in from the oneRingList of the other one node.
      Important because different Point-orders cause different currents."'
00070     partnerElement =0
00071     for i in triangle:
00072         if i != nodeInd2:
00073             differentElement = i
00074     for j in trianglesWithBothNodes[1]:
00075         if differentElement in j:
00076             partnerElement = j
00077     return partnerElement
00078
00079 def elementInArray(array,value):
00080     "'returns a list with the elements the contain "value""'
00081     solution =[]
00082     for element in array:
00083         if value in element:
00084             solution.append(element)
00085     return solution
00086
00087 def elementInAandB(a, b):
00088     "'returns a list with elements that are in a and b"'
00089     solution =[]
00090     for i in a:
00091         if i in b: solution.append(i)
00092     return solution
00093
00094 def compareMultipleElementsBoolean(elements,testelements):
00095     "'returns a array with boolean elements in the length of elements, True if the value is in
      testelements False otherwise"'
00096     comparisonResult=[]
00097     for i in elements:
00098         if i == testelements:
00099             comparisonResult.append(True)
00100         else:
00101             comparisonResult.append(False)
00102     return comparisonResult
```

```
00103
00104 def getSpatialDistancesMatrix(mesh):
00105     "'returns a matrix containing the spatial distance between node i and node j"'
00106     nodalNeighbourMatrix = np.full((len(mesh.vertices),len(mesh.vertices)), 0,dtype=float)
00107     for i in range(len(mesh.vertices)):
00108         for j in range(len(mesh.vertices)):
00109             nodalNeighbourMatrix[i][j] = distanceBetweenPoints(mesh.vertices[i],mesh.vertices[j])
00110     return nodalNeighbourMatrix
00111
00112 def getNeighbourhoodMatrix(mesh):
00113     "'returns a boolean matrix with information if node i and node j are neighbours"'
00114     nodeAdjacencyMatrix = np.full((len(mesh.vertices),len(mesh.vertices) ), False)
00115     for i in range(len(mesh.faces)):
00116         nodeAdjacencyMatrix[mesh.faces[i][0]][mesh.faces[i][1]]=True
00117         nodeAdjacencyMatrix[mesh.faces[i][1]][mesh.faces[i][2]]=True
00118         nodeAdjacencyMatrix[mesh.faces[i][2]][mesh.faces[i][0]]=True
00119     return nodeAdjacencyMatrix
00120
00121
00122
00123 def compareMultipleElementsBooleanTest():
00124     "'Test function: should always be True"'
00125     return len(np.where(compareMultipleElementsBoolean()))==1
```

8.16 subfunctions/sensitivityMatrix.py File Reference

Namespaces

- namespace sensitivityMatrix

Functions

- def sensitivityMatrix.getSensitivityMatrix (test, mesh, target, n)
- def sensitivityMatrix.calcSensitivityMat (mesh, biotSavatCoeff, target, u, v, gaussWeight)
- def sensitivityMatrix.gaussLegendreIntegrationPointsTriangle (test, n)
- def sensitivityMatrix.calcWeightsGauss (n)

8.17 sensitivityMatrix.py

Go to the documentation of this file.

```
00001 #Flächenströme der Dreieck um jeden Knoten aufintegrieren
00002 import numpy as np
00003 import scipy
00004
00005 def getSensitivityMatrix(test,mesh,target,n):
00006     "'returns the sensitivity Matrix for the mesh"'
00007     biotSavatCoeff = 10**(-7)
00008     [u,v,gaussWeight] = gaussLegendreIntegrationPointsTriangle(test,n)
00009     sensitivityMatrix = calcSensitivityMat(mesh,biotSavatCoeff,target,u,v,gaussWeight)
00010     return sensitivityMatrix
00011
00012 def calcSensitivityMat(mesh,biotSavatCoeff,target,u,v,gaussWeight):
00013     "'returns the calculated sensitivityMatrix"'
00014     trianglesPerNode = [len(mesh.neighbours[x]) for x in range(len(mesh.neighbours))]
00015     xTarget,yTarget,zTarget = target.vertices[:,0],target.vertices[:,1],target.vertices[:,2]
00016     xAll,yAll,zAll=[],[],[]
00017     for nodeIndex in range(len(mesh.vertices)):
00018         dCx,dCy,dCz = np.zeros(len(xTarget)),np.zeros(len(xTarget)),np.zeros(len(xTarget))
00019         for triangleIndex in range(trianglesPerNode[nodeIndex]):
00020             nodePoint = mesh.vertices[nodeIndex]
00021             pointB = mesh.vertices[mesh.oneRingList[nodeIndex][triangleIndex][0]]
00022             pointC = mesh.vertices[mesh.oneRingList[nodeIndex][triangleIndex][1]]
00023             nodeX,nodeY,nodeZ = nodePoint
00024             bX,bY,bZ = pointB
00025             cX,cY,cZ = pointC
00026             vX,vY,vZ = (pointC -
      pointB)/(scipy.linalg.norm(np.cross(pointC-nodePoint,pointB-nodePoint)))
00027             for gaussIndex in range(len(gaussWeight)):
```

```
00028                 xGaussInUV =
      nodeX*u[gaussIndex]+bX*v[gaussIndex]+cX*(1-u[gaussIndex]-v[gaussIndex])#scalar
00029                 yGaussInUV =
      nodeY*u[gaussIndex]+bY*v[gaussIndex]+cY*(1-u[gaussIndex]-v[gaussIndex])#scalar
00030                 zGaussInUV =
      nodeZ*u[gaussIndex]+bZ*v[gaussIndex]+cZ*(1-u[gaussIndex]-v[gaussIndex])#scalar
00031                 distanceNorm =
      (np.square(xGaussInUV-xTarget)+np.square(yGaussInUV-yTarget)+np.square(zGaussInUV-zTarget))**(-3/2)#for
      biot savat #len of target
00032                 dCx = dCx + ((-1)*vZ*(yTarget-yGaussInUV)+ vY*(zTarget-zGaussInUV))*distanceNorm *2
      *mesh.areas[mesh.neighbours[nodeIndex][triangleIndex]]* gaussWeight[gaussIndex]
00033                 dCy = dCy + ((-1)*vX*(zTarget-zGaussInUV)+ vZ*(xTarget-xGaussInUV))*distanceNorm *2
      *mesh.areas[mesh.neighbours[nodeIndex][triangleIndex]]* gaussWeight[gaussIndex]
00034                 dCz = dCz + ((-1)*vY*(xTarget-xGaussInUV)+ vX*(yTarget-yGaussInUV))*distanceNorm *2
      *mesh.areas[mesh.neighbours[nodeIndex][triangleIndex]]* gaussWeight[gaussIndex]
00035         dCx *= biotSavatCoeff
00036         dCy *= biotSavatCoeff
00037         dCz *= biotSavatCoeff
00038         xAll.append(dCx)
00039         yAll.append(dCy)
00040         zAll.append(dCz)
00041     return [xAll,yAll,zAll]
00042
00043 def gaussLegendreIntegrationPointsTriangle(test,n):
00044     "'returns the weights and the test point for the gauss legendre"'
00045     u,v,ck=[],[],[]
00046     eta,w = calcWeightsGauss(n)
00047     test.calcWeightsGauss = eta,w
00048     for i in range(len(eta)):
00049         for j in range(len(eta)):
00050             u.append((1+eta[i])/2)
00051             v.append((1-eta[i])*(1+eta[j])/4)
00052             ck.append(((1-eta[i])/8)*w[i]*w[j])
00053     test.gaußLegendre = [u,v,ck]
00054     return [u,v,ck]
00055
00056 def calcWeightsGauss(n):
00057     "'returns the abscissa and the weights for a Gauss-Legendre quadrature"'
00058     abscissa = np.zeros(n)
00059     weights = np.copy(abscissa)
00060     m = (n+1)/2
00061     for i in np.arange(1,m).reshape(-1):
00062         z = np.cos(np.pi*(i-0.25)/(n+0.5))
00063         z1 = z+1
00064         while abs(z-z1)>(2.2204*10**(-16)):#distance from 1.0 to the next larger double precision
      number
00065             p1 = 1
00066             p2 = 0
00067             for j in range(n):
00068                 p3 = np.copy(p2)
00069                 p2 = np.copy(p1)
00070                 p1 = ((2*(j+1)-1)*z*p2-((j+1)-1)*p3)/(j+1)
00071             pp = n*(z*p1-p2)/(z**2-1)
00072             z1 = np.copy(z)
00073             z = z1-p1/pp
00074         abscissa[int(i-1)] = -z
00075         abscissa[int(n+1-i-1)] = z
00076         weights[int(i-1)] = 2/((1-z**2)*pp**2)
00077         weights[int(n+1-i-1)] = weights[int(i-1)]
00078     return abscissa,weights
```

8.18 subfunctions/streamFunctionOptimization.py File Reference

Namespaces

- namespace streamFunctionOptimization

Functions

- def streamFunctionOptimization.streamFunctionOptimization (test, mesh, target, sensitivityMatrix, resistanceMatrix, tikonovFactor)
- def streamFunctionOptimization.applyTikonovRegularisation (tikonovFactor, redSenMat, redResMat, target)
- def streamFunctionOptimization.reduceMatricesForBoundaryNodes (mesh, matToRed, zeroFlag)
- def streamFunctionOptimization.rearrangeReducedMat (mesh, dimToRed, reducedMat)

- def streamFunctionOptimization.getBoundaryDetails (mesh)
- def streamFunctionOptimization.getReducedMat (mesh, dimToRed, reducedMat, zeroFlag)
- def streamFunctionOptimization.getNotBoundaryNodes (mesh)
- def streamFunctionOptimization.getDimToRed (matToRed)
- def streamFunctionOptimization.getNumNodesPerBoundary (mesh)
- def streamFunctionOptimization.reexpandSteamFunctionForBoundaryNodes (mesh, reducedSF, boundary↩ Nodes, isNotBoundaryNode, zeroFlag)
- def streamFunctionOptimization.updateMeshCurrentDensityMeshFaces (mesh, optStreamFkt)

8.19 streamFunctionOptimization.py

Go to the documentation of this file.

```
00001 import numpy as np
00002 import scipy.linalg as sc
00003
00004 def streamFunctionOptimization(test,mesh,target,sensitivityMatrix,resistanceMatrix,tikonovFactor):
00005     "'returns the magnetic field generated by the optimized stream function. Version for only one coil
      part, else  a combined mesh is needed"'
00006     PotentialZeroAtBoundaryNodes = False
00007     sensitivityMatrixSingleZKomp=np.transpose(np.array(sensitivityMatrix)[2])
00008
00009     redResMat,boundaryNodes,isNotBoundaryNode =
      reduceMatricesForBoundaryNodes(mesh,resistanceMatrix,PotentialZeroAtBoundaryNodes)
00010     redSenMat,boundaryNodes,isNotBoundaryNode =
      reduceMatricesForBoundaryNodes(mesh,sensitivityMatrixSingleZKomp,PotentialZeroAtBoundaryNodes)
00011
00012     reducedSF = applyTikonovRegularisation(tikonovFactor,redSenMat,redResMat, target)
00013     test.reducedSF = reducedSF
00014
00015     if not PotentialZeroAtBoundaryNodes:
00016         optStreamFkt =
      reexpandSteamFunctionForBoundaryNodes(mesh,reducedSF,boundaryNodes,isNotBoundaryNode,PotentialZeroAtBoundaryNodes)
00017     else: optStreamFkt = reducedSF
00018
00019     bFieldGeneratedByOptSF = [np.dot(np.transpose(sensitivityMatrix[0]),optStreamFkt),
      np.dot(np.transpose(sensitivityMatrix[1]),optStreamFkt),
      np.dot(np.transpose(sensitivityMatrix[2]),optStreamFkt)]
00020
00021     updateMeshCurrentDensityMeshFaces(mesh,optStreamFkt)
00022     return bFieldGeneratedByOptSF,optStreamFkt
00023
00024 def applyTikonovRegularisation(tikonovFactor,redSenMat,redResMat, target):
00025     "'returns the StreamFunction for the reduced Matrix with applied trikonov regularisation."'
00026     tikonovFactor = tikonovFactor*np.shape(redSenMat)[0]/np.shape(redSenMat)[1]
00027     tikRegMat = tikonovFactor * redResMat
00028     reducedSF = np.dot(sc.pinv(np.dot(np.transpose(np.array(redSenMat)),redSenMat) +
      np.dot(np.transpose(np.array(tikRegMat)),tikRegMat)),np.dot(np.transpose(np.array(redSenMat)),np.transpose(target.field
00029     return reducedSF
00030
00031 def reduceMatricesForBoundaryNodes(mesh, matToRed,zeroFlag):
00032     "'returns the for the boundary nodes reduced matrix"'
00033     dimToRed = getDimToRed(matToRed)
00034     if True in dimToRed:
00035         reducedMat = getReducedMat(mesh,dimToRed,matToRed,zeroFlag)
00036         reducedMat,boundaryNodes,notBoundaryNodes = rearrangeReducedMat(mesh,dimToRed,reducedMat)
00037     else:
00038         print("nothing to reduce")
00039     return [reducedMat,boundaryNodes,notBoundaryNodes]
00040
00041 def rearrangeReducedMat(mesh,dimToRed,reducedMat):
00042     "'rearranges the reduced matrix. returns: reducedMat,boundaryNodes,notBoundaryNodes"'
00043     numNodesPerBoundary,notBoundaryNodes,boundaryNodes = getBoundaryDetails(mesh)
00044     boundaryNodesFirstInds = [boundaryNodes[i][0] for i in range(len(mesh.openBoundaries))]
00045     for dimToRedInd in np.nonzero(dimToRed)[0]:
00046         prevReducedMat = np.copy(reducedMat)
00047         index1=np.array([':']*np.ndim(prevReducedMat),dtype=object)
00048         index2,index3,index4,index5 = np.copy(index1),np.copy(index1),np.copy(index1),np.copy(index1)
00049         index1[dimToRedInd] = [i for i in range(len(mesh.openBoundaries))]
00050         index2[dimToRedInd] = [x for x in boundaryNodesFirstInds]
00051         index3[dimToRedInd] =
      np.arange((len(mesh.openBoundaries)),(len(notBoundaryNodes)+len(mesh.openBoundaries)))
00052         index4[dimToRedInd] = np.transpose(notBoundaryNodes)
00053         index5[dimToRedInd] =
      np.arange(((len(prevReducedMat[0])-(sum(numNodesPerBoundary)-len(mesh.openBoundaries)))),
      len(prevReducedMat[0]))
00054
00055
```

```
00056         if index1[0] == ':' and index2[0]==':' and index3[0] == ':' and index4[0]==':' and index5[0]
      == ':':
00057             reducedMat[:,index1[1]] = prevReducedMat[:,index2[1]]
00058             reducedMat[:,index3[1]] = prevReducedMat[:,index4[1]]
00059             reducedMat=np.delete(reducedMat,index5[1],1)
00060         else:
00061             reducedMat[index1[0]] = prevReducedMat[index2[0]]
00062             reducedMat[index3[0]] = prevReducedMat[index4[0]]
00063             reducedMat=np.delete(reducedMat,index5[0],0)
00064     return reducedMat,boundaryNodes,notBoundaryNodes
00065
00066 def getBoundaryDetails(mesh):
00067     "'returns numNodesPerBoundary,notBoundaryNodes,boundaryNodes for the given mesh."'
00068     numNodesPerBoundary = getNumNodesPerBoundary(mesh)
00069     notBoundaryNodes = getNotBoundaryNodes(mesh)
00070     boundaryNodes = [mesh.openBoundaries[1],mesh.openBoundaries[0]]
00071     return numNodesPerBoundary,notBoundaryNodes,boundaryNodes
00072
00073 def getReducedMat(mesh,dimToRed,reducedMat,zeroFlag):
00074     "'returns the reduced matrix"'
00075     boundaryNodes = [mesh.openBoundaries[1],mesh.openBoundaries[0]]
00076     for dimToRedInd in np.nonzero(dimToRed)[0]:
00077             index1=np.array([':']*np.ndim(reducedMat),dtype=object)
00078             index2=np.copy(index1)
00079             for boundaryInd in range(len(mesh.openBoundaries)):
00080                 if zeroFlag:
00081                     index1[dimToRedInd] = boundaryNodes[boundaryInd][0]
00082                     reducedMat[index1[0:None]]=0
00083                 else:
00084                     index1[dimToRedInd] = boundaryNodes[boundaryInd][0]
00085                     index2[dimToRedInd] = np.array(boundaryNodes[boundaryInd][0:None],dtype=int)
00086                     if dimToRedInd:#(==1)
00087                         reducedMat[:,index1[1]] =
      np.sum(np.array(reducedMat)[:,index2[1]],dimToRedInd)#spalte ersetzen
00088                     else:
00089                         reducedMat[index1[0]] = np.sum(np.array(reducedMat)[index2[0]],dimToRedInd)
00090     return reducedMat
00091
00092 def getNotBoundaryNodes(mesh):
00093     "'returns no-boundary nodes of a given mesh"'
00094     isNotBoundaryNode = []
00095     for i in range(len(mesh.vertices[:,0])):
00096         if mesh.boundary[i]:
00097             continue
00098         else: isNotBoundaryNode.append(i)
00099     return isNotBoundaryNode
00100
00101 def getDimToRed(matToRed):
00102     "'returns boolean values which dimension should be reduced"'
00103     dimToRed=[]
00104     for i in np.shape(matToRed):
00105         if i == len(matToRed[0]): dimToRed.append(True)
00106         else: dimToRed.append(False)
00107     return dimToRed
00108
00109 def getNumNodesPerBoundary(mesh):
00110     "'returns the number of nodes for every boundary"'
00111     numNodesPerBoundary=[]
00112     for i in range(len(mesh.openBoundaries)):
00113         numNodesPerBoundary.append(len(mesh.openBoundaries[i]))
00114     return numNodesPerBoundary
00115
00116 def reexpandSteamFunctionForBoundaryNodes(mesh,reducedSF,boundaryNodes,isNotBoundaryNode,zeroFlag):
00117     "'reexpand Streamfunction to all nodes, because the nodes of the first boundary have a potential
      of zero"'
00118     streamFunction = np.zeros(len(mesh.vertices))
00119     for boundaryInd in range(len(boundaryNodes)):
00120         if zeroFlag:
00121             streamFunction[boundaryNodes[boundaryInd]]=0
00122         else:
00123             streamFunction[boundaryNodes[boundaryInd]]=reducedSF[boundaryInd]
00124
00125     streamFunction[isNotBoundaryNode]=reducedSF[len(boundaryNodes):]
00126     return streamFunction
00127
00128 def updateMeshCurrentDensityMeshFaces(mesh,optStreamFkt):
00129     "'updates the current density of the Faces in the mesh"'
00130     pot1 =
      np.transpose(np.array([optStreamFkt[mesh.faces[:,2]]-optStreamFkt[mesh.faces[:,0]],optStreamFkt[mesh.faces[:,2]]-optSt
00131     pot2 =
      np.transpose(np.array([optStreamFkt[mesh.faces[:,1]]-optStreamFkt[mesh.faces[:,0]],optStreamFkt[mesh.faces[:,1]]-optSt
00132     pot3 =
      np.transpose(np.array([optStreamFkt[mesh.faces[:,2]]-optStreamFkt[mesh.faces[:,1]],optStreamFkt[mesh.faces[:,2]]-optSt
00133     edge1 = mesh.vertices[mesh.faces[:,2]]-mesh.vertices[mesh.faces[:,0]]
00134     edge2 = mesh.vertices[mesh.faces[:,1]]-mesh.vertices[mesh.faces[:,0]]
00135     edge3 = mesh.vertices[mesh.faces[:,2]]-mesh.vertices[mesh.faces[:,1]]
00136     mesh.currentDensityFaces = edge1*pot1+edge2*pot2+edge3*pot3
```

00137

8.20 subfunctions/testCase.py File Reference

Namespaces

- namespace testCase

Functions

- def testCase.test_finalSF ()
- def testCase.test_bFieldGeneratedByOptSF ()
- def testCase.test_reducedSF ()
- def testCase.test_gaußLegendre ()
- def testCase.test_WeightsGauss ()
- def testCase.test_matElementsShouldGetValue ()
- def testCase.main ()

Variables

- string testCase.meshFile = "cylinder_radius500mm_length1500mm.stl"
- string testCase.targetMeshFile = "sphere_radius150mm.stl"
- int testCase.gaussOrder = 2
- int testCase.tikonovFac = 100
- float testCase.specificConductivityMaterial = 1.8000∗10∗∗-8
- float testCase.conducterThickness = 0.005
- float testCase.materialFactor = specificConductivityMaterial/conducterThickness
- testCase.Test = Tester()
- testCase.Mesh = CylindricMeshGiven(meshFile)
- testCase.TargetSphere = TargetFieldGiven(targetMeshFile,1)
- testCase.sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
- testCase.resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
- testCase.BField
- testCase.SFOpt
- list testCase.matElementsShouldGetValueCorrect = [0.0000000000, 1.0000000000, 2.0000000000, 3.↩ 0000000000, 4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000, 9.0000000000, 10.0000000000, 11.0000000000, 12.0000000000, 13.0000000000, 14.0000000000, 15.0000000000, 16.↩ 0000000000, 17.0000000000, 18.0000000000, 19.0000000000, 20.0000000000, 21.0000000000, 22.↩ 0000000000, 23.0000000000, 24.0000000000, 25.0000000000, 26.0000000000, 27.0000000000, 28.↩ 0000000000, 29.0000000000, 30.0000000000, 31.0000000000, 32.0000000000, 33.0000000000, 34.↩ 0000000000, 35.0000000000, 36.0000000000, 37.0000000000, 38.0000000000, 39.0000000000, 40.↩ 0000000000, 41.0000000000, 42.0000000000, 43.0000000000, 44.0000000000, 45.0000000000, 46.↩ 0000000000, 47.0000000000, 48.0000000000, 49.0000000000, 50.0000000000, 51.0000000000, 52.↩ 0000000000, 53.0000000000, 54.0000000000, 55.0000000000, 56.0000000000, 57.0000000000, 58.↩ 0000000000, 59.0000000000, 60.0000000000, 61.0000000000, 62.0000000000, 63.0000000000, 64.↩ 0000000000, 65.0000000000, 66.0000000000, 67.0000000000, 68.0000000000, 69.0000000000, 70.↩ 0000000000, 71.0000000000, 72.0000000000, 73.0000000000, 74.0000000000, 75.0000000000, 76.↩ 0000000000, 77.0000000000, 78.0000000000, 79.0000000000, 80.0000000000, 81.0000000000, 82.↩ 0000000000, 83.0000000000, 84.0000000000, 85.0000000000, 86.0000000000, 87.0000000000, 88.↩ 0000000000, 89.0000000000, 90.0000000000, 91.0000000000, 92.0000000000, 93.0000000000, 94.↩ 0000000000, 95.0000000000, 96.0000000000, 97.0000000000, 98.0000000000, 99.0000000000, 100.↩ 0000000000, 101.0000000000, 102.0000000000, 103.0000000000, 104.0000000000, 105.0000000000,

106.0000000000, 107.0000000000, 108.0000000000, 109.0000000000, 110.0000000000, 111.↩
0000000000, 112.0000000000, 113.0000000000, 114.0000000000, 115.0000000000, 116.0000000000,
117.0000000000, 118.0000000000, 119.0000000000, 120.0000000000, 121.0000000000, 122.↩
0000000000, 123.0000000000, 124.0000000000, 125.0000000000, 126.0000000000, 127.0000000000,
128.0000000000, 129.0000000000, 130.0000000000, 131.0000000000, 132.0000000000, 133.↩
0000000000, 134.0000000000, 135.0000000000, 136.0000000000, 137.0000000000, 138.0000000000,
139.0000000000, 140.0000000000, 141.0000000000, 142.0000000000, 143.0000000000, 144.↩
0000000000, 145.0000000000, 146.0000000000, 147.0000000000, 148.0000000000, 149.0000000000,
150.0000000000, 151.0000000000, 152.0000000000, 153.0000000000, 154.0000000000, 155.↩
0000000000, 156.0000000000, 157.0000000000, 158.0000000000, 159.0000000000, 160.0000000000,
161.0000000000, 162.0000000000, 163.0000000000, 164.0000000000, 165.0000000000, 166.↩
0000000000, 167.0000000000, 168.0000000000, 169.0000000000, 170.0000000000, 171.0000000000,
172.0000000000, 173.0000000000, 174.0000000000, 175.0000000000, 176.0000000000, 177.↩
0000000000, 178.0000000000, 179.0000000000, 180.0000000000, 181.0000000000, 182.0000000000,
183.0000000000, 184.0000000000, 185.0000000000, 186.0000000000, 187.0000000000, 188.↩
0000000000, 189.0000000000, 190.0000000000, 191.0000000000, 192.0000000000, 193.0000000000,
194.0000000000, 195.0000000000, 196.0000000000, 197.0000000000, 198.0000000000, 199.↩
0000000000, 200.0000000000, 201.0000000000, 202.0000000000, 203.0000000000, 204.0000000000,
205.0000000000, 206.0000000000, 207.0000000000, 208.0000000000, 209.0000000000, 210.↩
0000000000, 211.0000000000, 212.0000000000, 213.0000000000, 214.0000000000, 215.0000000000,
216.0000000000, 217.0000000000, 218.0000000000, 219.0000000000, 220.0000000000, 221.↩
0000000000, 222.0000000000, 223.0000000000, 224.0000000000, 225.0000000000, 226.0000000000,
227.0000000000, 228.0000000000, 229.0000000000, 230.0000000000, 231.0000000000, 232.↩
0000000000, 233.0000000000, 234.0000000000, 235.0000000000, 236.0000000000, 237.0000000000,
238.0000000000, 239.0000000000, 240.0000000000, 241.0000000000, 242.0000000000, 243.↩
0000000000, 244.0000000000, 245.0000000000, 246.0000000000, 247.0000000000, 248.0000000000,
249.0000000000, 250.0000000000, 251.0000000000, 252.0000000000, 253.0000000000, 254.↩
0000000000, 255.0000000000, 256.0000000000, 257.0000000000, 258.0000000000, 259.0000000000,
260.0000000000, 261.0000000000, 262.0000000000, 263.0000000000, 1.0000000000, 2.0000000000,
0.0000000000, 6.0000000000, 8.0000000000, 9.0000000000, 3.0000000000, 2.0000000000, 7.↩
0000000000, 3.0000000000, 1.0000000000, 5.0000000000, 4.0000000000, 1.0000000000, 9.0000000000,
4.0000000000, 2.0000000000, 12.0000000000, 10.0000000000, 18.0000000000, 12.0000000000, 2.↩
0000000000, 5.0000000000, 3.0000000000, 10.0000000000, 4.0000000000, 11.0000000000, 14.↩
0000000000, 8.0000000000, 15.0000000000, 7.0000000000, 1.0000000000, 14.0000000000, 6.↩
0000000000, 0.0000000000, 1.0000000000, 6.0000000000, 15.0000000000, 16.0000000000, 17.↩
0000000000, 1.0000000000, 23.0000000000, 9.0000000000, 12.0000000000, 17.0000000000, 1.↩
0000000000, 3.0000000000, 8.0000000000, 18.0000000000, 78.0000000000, 4.0000000000, 5.↩
0000000000, 81.0000000000, 77.0000000000, 11.0000000000, 77.0000000000, 10.0000000000, 18.↩
0000000000, 9.0000000000, 17.0000000000, 4.0000000000, 3.0000000000, 19.0000000000, 21.↩
0000000000, 7.0000000000, 14.0000000000, 15.0000000000, 22.0000000000, 21.0000000000, 13.↩
0000000000, 7.0000000000, 6.0000000000, 14.0000000000, 29.0000000000, 8.0000000000, 6.↩
0000000000, 22.0000000000, 23.0000000000, 8.0000000000, 30.0000000000, 17.0000000000, 23.↩
0000000000, 25.0000000000, 24.0000000000, 9.0000000000, 12.0000000000, 25.0000000000, 16.↩
0000000000, 27.0000000000, 8.0000000000, 19.0000000000, 26.0000000000, 12.0000000000, 10.↩
0000000000, 4.0000000000, 81.0000000000, 85.0000000000, 19.0000000000, 26.0000000000, 33.↩
0000000000, 27.0000000000, 12.0000000000, 18.0000000000, 17.0000000000, 21.0000000000, 13.↩
0000000000, 14.0000000000, 20.0000000000, 28.0000000000, 13.0000000000, 144.0000000000, 143.↩
0000000000, 22.0000000000, 15.0000000000, 28.0000000000, 14.0000000000, 35.0000000000, 29.↩
0000000000, 21.0000000000, 29.0000000000, 30.0000000000, 8.0000000000, 15.0000000000, 16.↩
0000000000, 37.0000000000, 30.0000000000, 31.0000000000, 25.0000000000, 16.0000000000, 32.↩
0000000000, 16.0000000000, 17.0000000000, 31.0000000000, 27.0000000000, 24.0000000000, 85.↩
0000000000, 18.0000000000, 19.0000000000, 33.0000000000, 90.0000000000, 33.0000000000, 32.↩
0000000000, 17.0000000000, 25.0000000000, 19.0000000000, 34.0000000000, 151.0000000000, 22.↩
0000000000, 35.0000000000, 147.0000000000, 144.0000000000, 21.0000000000, 30.0000000000, 35.↩
0000000000, 36.0000000000, 22.0000000000, 23.0000000000, 15.0000000000, 29.0000000000, 43.↩
0000000000, 37.0000000000, 23.0000000000, 24.0000000000, 36.0000000000, 16.0000000000, 37.↩
0000000000, 24.0000000000, 32.0000000000, 39.0000000000, 38.0000000000, 25.0000000000, 39.↩
0000000000, 41.0000000000, 31.0000000000, 25.0000000000, 34.0000000000, 27.0000000000, 19.↩

0000000000, 27.0000000000, 34.0000000000, 90.0000000000, 40.0000000000, 26.0000000000, 41.↩
0000000000, 40.0000000000, 47.0000000000, 32.0000000000, 27.0000000000, 33.0000000000, 36.↩
0000000000, 29.0000000000, 151.0000000000, 22.0000000000, 42.0000000000, 28.0000000000, 42.↩
0000000000, 29.0000000000, 35.0000000000, 30.0000000000, 43.0000000000, 43.0000000000, 44.↩
0000000000, 24.0000000000, 30.0000000000, 31.0000000000, 38.0000000000, 31.0000000000, 39.↩
0000000000, 37.0000000000, 46.0000000000, 45.0000000000, 44.0000000000, 46.0000000000, 48.↩
0000000000, 41.0000000000, 38.0000000000, 31.0000000000, 32.0000000000, 34.0000000000, 90.↩
0000000000, 33.0000000000, 96.0000000000, 47.0000000000, 103.0000000000, 34.0000000000, 54.↩
0000000000, 47.0000000000, 48.0000000000, 32.0000000000, 39.0000000000, 35.0000000000, 43.↩
0000000000, 156.0000000000, 36.0000000000, 162.0000000000, 49.0000000000, 151.0000000000, 50.↩
0000000000, 36.0000000000, 37.0000000000, 56.0000000000, 44.0000000000, 30.0000000000, 42.↩
0000000000, 49.0000000000, 51.0000000000, 38.0000000000, 43.0000000000, 57.0000000000, 45.↩
0000000000, 37.0000000000, 50.0000000000, 52.0000000000, 51.0000000000, 46.0000000000, 44.↩
0000000000, 38.0000000000, 39.0000000000, 38.0000000000, 52.0000000000, 53.0000000000, 45.↩
0000000000, 48.0000000000, 34.0000000000, 103.0000000000, 40.0000000000, 54.0000000000, 41.↩
0000000000, 39.0000000000, 53.0000000000, 55.0000000000, 41.0000000000, 54.0000000000, 46.↩
0000000000, 61.0000000000, 42.0000000000, 162.0000000000, 43.0000000000, 56.0000000000, 57.↩
0000000000, 63.0000000000, 56.0000000000, 44.0000000000, 43.0000000000, 45.0000000000, 44.↩
0000000000, 52.0000000000, 58.0000000000, 57.0000000000, 65.0000000000, 45.0000000000, 46.↩
0000000000, 51.0000000000, 60.0000000000, 59.0000000000, 58.0000000000, 53.0000000000, 46.↩
0000000000, 55.0000000000, 52.0000000000, 60.0000000000, 48.0000000000, 48.0000000000, 109.↩
0000000000, 61.0000000000, 47.0000000000, 103.0000000000, 115.0000000000, 41.0000000000, 60.↩
0000000000, 62.0000000000, 61.0000000000, 48.0000000000, 53.0000000000, 50.0000000000, 49.↩
0000000000, 43.0000000000, 63.0000000000, 169.0000000000, 162.0000000000, 58.0000000000, 51.↩
0000000000, 63.0000000000, 64.0000000000, 50.0000000000, 44.0000000000, 51.0000000000, 57.↩
0000000000, 65.0000000000, 52.0000000000, 64.0000000000, 67.0000000000, 72.0000000000, 52.↩
0000000000, 66.0000000000, 65.0000000000, 60.0000000000, 52.0000000000, 53.0000000000, 59.↩
0000000000, 67.0000000000, 62.0000000000, 55.0000000000, 121.0000000000, 68.0000000000, 62.↩
0000000000, 115.0000000000, 48.0000000000, 55.0000000000, 54.0000000000, 55.0000000000, 68.↩
0000000000, 61.0000000000, 60.0000000000, 69.0000000000, 67.0000000000, 64.0000000000, 56.↩
0000000000, 57.0000000000, 50.0000000000, 175.0000000000, 70.0000000000, 169.0000000000, 58.↩
0000000000, 57.0000000000, 71.0000000000, 70.0000000000, 63.0000000000, 65.0000000000, 52.↩
0000000000, 64.0000000000, 58.0000000000, 59.0000000000, 72.0000000000, 75.0000000000, 71.↩
0000000000, 59.0000000000, 72.0000000000, 66.0000000000, 62.0000000000, 59.0000000000, 60.↩
0000000000, 127.0000000000, 69.0000000000, 73.0000000000, 121.0000000000, 61.0000000000, 62.↩
0000000000, 68.0000000000, 67.0000000000, 62.0000000000, 63.0000000000, 71.0000000000, 64.↩
0000000000, 181.0000000000, 187.0000000000, 175.0000000000, 74.0000000000, 74.0000000000,
64.0000000000, 70.0000000000, 65.0000000000, 75.0000000000, 75.0000000000, 65.0000000000,
59.0000000000, 68.0000000000, 69.0000000000, 76.0000000000, 75.0000000000, 71.0000000000,
187.0000000000, 193.0000000000, 70.0000000000, 74.0000000000, 65.0000000000, 76.0000000000,
71.0000000000, 193.0000000000, 74.0000000000, 10.0000000000, 78.0000000000, 79.0000000000,
10.0000000000, 84.0000000000, 77.0000000000, 81.0000000000, 80.0000000000, 79.0000000000, 83.↩
0000000000, 80.0000000000, 82.0000000000, 78.0000000000, 79.0000000000, 82.0000000000, 84.↩
0000000000, 78.0000000000, 18.0000000000, 10.0000000000, 84.0000000000, 85.0000000000, 78.↩
0000000000, 88.0000000000, 86.0000000000, 80.0000000000, 83.0000000000, 87.0000000000, 84.↩
0000000000, 79.0000000000, 86.0000000000, 82.0000000000, 82.0000000000, 81.0000000000, 89.↩
0000000000, 88.0000000000, 80.0000000000, 85.0000000000, 78.0000000000, 81.0000000000, 18.↩
0000000000, 84.0000000000, 26.0000000000, 90.0000000000, 89.0000000000, 91.0000000000, 87.↩
0000000000, 92.0000000000, 82.0000000000, 93.0000000000, 82.0000000000, 91.0000000000, 88.↩
0000000000, 86.0000000000, 89.0000000000, 93.0000000000, 87.0000000000, 82.0000000000, 94.↩
0000000000, 84.0000000000, 90.0000000000, 84.0000000000, 88.0000000000, 95.0000000000, 85.↩
0000000000, 94.0000000000, 89.0000000000, 40.0000000000, 102.0000000000, 26.0000000000, 33.↩
0000000000, 95.0000000000, 85.0000000000, 96.0000000000, 93.0000000000, 92.0000000000, 97.↩
0000000000, 99.0000000000, 86.0000000000, 87.0000000000, 98.0000000000, 91.0000000000, 97.↩
0000000000, 87.0000000000, 99.0000000000, 100.0000000000, 91.0000000000, 94.0000000000, 88.↩
0000000000, 105.0000000000, 89.0000000000, 88.0000000000, 106.0000000000, 100.0000000000,
101.0000000000, 93.0000000000, 95.0000000000, 101.0000000000, 102.0000000000, 90.0000000000,
89.0000000000, 94.0000000000, 103.0000000000, 108.0000000000, 40.0000000000, 90.0000000000,

102.0000000000, 209.0000000000, 210.0000000000, 104.0000000000, 92.0000000000, 213.0000000000, 99.0000000000, 98.0000000000, 91.0000000000, 97.0000000000, 209.0000000000, 104.0000000000, 105.0000000000, 93.0000000000, 97.0000000000, 91.0000000000, 94.0000000000, 93.0000000000, 105.0000000000, 106.0000000000, 102.0000000000, 106.0000000000, 94.0000000000, 95.0000000000, 112.0000000000, 107.0000000000, 101.0000000000, 107.0000000000, 96.0000000000, 113.0000000000, 95.0000000000, 90.0000000000, 108.0000000000, 108.0000000000, 109.0000000000, 114.0000000000, 47.0000000000, 96.0000000000, 40.0000000000, 54.0000000000, 110.0000000000, 99.0000000000, 213.0000000000, 97.0000000000, 219.0000000000, 105.0000000000, 100.0000000000, 106.0000000000, 99.0000000000, 104.0000000000, 93.0000000000, 110.0000000000, 116.0000000000, 111.0000000000, 105.0000000000, 100.0000000000, 112.0000000000, 111.0000000000, 117.0000000000, 94.0000000000, 101.0000000000, 113.0000000000, 118.0000000000, 102.0000000000, 101.0000000000, 112.↩
0000000000, 103.0000000000, 113.0000000000, 114.0000000000, 102.0000000000, 96.0000000000, 114.0000000000, 115.0000000000, 103.0000000000, 54.0000000000, 120.0000000000, 116.0000000000, 219.0000000000, 105.0000000000, 225.0000000000, 104.0000000000, 105.0000000000, 116.↩
0000000000, 117.0000000000, 106.0000000000, 106.0000000000, 107.0000000000, 101.0000000000, 118.0000000000, 117.0000000000, 123.0000000000, 118.0000000000, 108.0000000000, 107.↩
0000000000, 114.0000000000, 102.0000000000, 119.0000000000, 113.0000000000, 120.0000000000, 108.0000000000, 109.0000000000, 119.0000000000, 103.0000000000, 120.0000000000, 121.↩
0000000000, 61.0000000000, 54.0000000000, 109.0000000000, 117.0000000000, 122.0000000000, 110.0000000000, 225.0000000000, 230.0000000000, 111.0000000000, 105.0000000000, 116.↩
0000000000, 123.0000000000, 122.0000000000, 111.0000000000, 112.0000000000, 106.0000000000, 123.0000000000, 107.0000000000, 112.0000000000, 119.0000000000, 124.0000000000, 113.↩
0000000000, 114.0000000000, 130.0000000000, 125.0000000000, 124.0000000000, 113.0000000000, 120.0000000000, 118.0000000000, 131.0000000000, 115.0000000000, 114.0000000000, 126.↩
0000000000, 125.0000000000, 121.0000000000, 109.0000000000, 119.0000000000, 61.0000000000, 132.0000000000, 115.0000000000, 126.0000000000, 120.0000000000, 127.0000000000, 68.0000000000, 123.0000000000, 116.0000000000, 230.0000000000, 235.0000000000, 117.0000000000, 128.↩
0000000000, 122.0000000000, 117.0000000000, 112.0000000000, 124.0000000000, 118.0000000000, 133.0000000000, 128.0000000000, 129.0000000000, 129.0000000000, 118.0000000000, 123.↩
0000000000, 130.0000000000, 119.0000000000, 130.0000000000, 120.0000000000, 119.0000000000, 131.0000000000, 131.0000000000, 121.0000000000, 120.0000000000, 132.0000000000, 73.0000000000, 68.0000000000, 121.0000000000, 133.0000000000, 235.0000000000, 240.0000000000, 122.0000000000, 123.0000000000, 130.0000000000, 123.0000000000, 133.0000000000, 137.0000000000, 124.↩
0000000000, 134.0000000000, 124.0000000000, 134.0000000000, 135.0000000000, 125.0000000000, 119.0000000000, 129.0000000000, 131.0000000000, 136.0000000000, 132.0000000000, 139.↩
0000000000, 135.0000000000, 130.0000000000, 125.0000000000, 120.0000000000, 126.0000000000, 126.0000000000, 121.0000000000, 131.0000000000, 127.0000000000, 240.0000000000, 128.↩
0000000000, 137.0000000000, 123.0000000000, 129.0000000000, 137.0000000000, 129.0000000000, 130.0000000000, 138.0000000000, 135.0000000000, 130.0000000000, 141.0000000000, 138.↩
0000000000, 134.0000000000, 139.0000000000, 131.0000000000, 131.0000000000, 132.0000000000, 138.0000000000, 140.0000000000, 240.0000000000, 129.0000000000, 133.0000000000, 134.↩
0000000000, 250.0000000000, 245.0000000000, 137.0000000000, 141.0000000000, 135.0000000000, 134.0000000000, 140.0000000000, 136.0000000000, 131.0000000000, 135.0000000000, 255.↩
0000000000, 138.0000000000, 141.0000000000, 142.0000000000, 137.0000000000, 250.0000000000, 139.0000000000, 140.0000000000, 135.0000000000, 138.0000000000, 140.0000000000, 141.↩
0000000000, 21.0000000000, 144.0000000000, 20.0000000000, 146.0000000000, 143.0000000000, 28.↩
0000000000, 21.0000000000, 146.0000000000, 147.0000000000, 146.0000000000, 149.0000000000, 143.0000000000, 149.0000000000, 145.0000000000, 144.0000000000, 150.0000000000, 147.↩
0000000000, 143.0000000000, 150.0000000000, 28.0000000000, 151.0000000000, 146.0000000000, 144.0000000000, 149.0000000000, 145.0000000000, 145.0000000000, 152.0000000000, 148.↩
0000000000, 153.0000000000, 154.0000000000, 146.0000000000, 150.0000000000, 154.0000000000, 155.0000000000, 147.0000000000, 149.0000000000, 151.0000000000, 160.0000000000, 146.↩
0000000000, 156.0000000000, 42.0000000000, 35.0000000000, 155.0000000000, 28.0000000000, 147.↩
0000000000, 150.0000000000, 148.0000000000, 149.0000000000, 153.0000000000, 157.0000000000, 158.0000000000, 159.0000000000, 154.0000000000, 152.0000000000, 149.0000000000, 149.↩
0000000000, 159.0000000000, 153.0000000000, 150.0000000000, 166.0000000000, 160.0000000000, 160.0000000000, 161.0000000000, 156.0000000000, 151.0000000000, 150.0000000000, 155.↩
0000000000, 162.0000000000, 161.0000000000, 151.0000000000, 168.0000000000, 42.0000000000,

164.0000000000, 153.0000000000, 158.0000000000, 152.0000000000, 157.0000000000, 159.↩
0000000000, 165.0000000000, 153.0000000000, 164.0000000000, 154.0000000000, 158.0000000000,
165.0000000000, 153.0000000000, 171.0000000000, 166.0000000000, 167.0000000000, 154.↩
0000000000, 155.0000000000, 150.0000000000, 166.0000000000, 161.0000000000, 155.0000000000,
168.0000000000, 167.0000000000, 156.0000000000, 173.0000000000, 160.0000000000, 49.0000000000,
42.0000000000, 168.0000000000, 56.0000000000, 156.0000000000, 169.0000000000, 164.0000000000,
157.0000000000, 165.0000000000, 163.0000000000, 214.0000000000, 215.0000000000, 170.↩
0000000000, 157.0000000000, 158.0000000000, 164.0000000000, 171.0000000000, 170.0000000000,
159.0000000000, 176.0000000000, 158.0000000000, 172.0000000000, 159.0000000000, 177.↩
0000000000, 171.0000000000, 154.0000000000, 167.0000000000, 160.0000000000, 173.0000000000,
172.0000000000, 178.0000000000, 166.0000000000, 160.0000000000, 161.0000000000, 162.↩
0000000000, 173.0000000000, 156.0000000000, 174.0000000000, 161.0000000000, 179.0000000000,
169.0000000000, 63.0000000000, 56.0000000000, 174.0000000000, 175.0000000000, 168.0000000000,
162.0000000000, 176.0000000000, 215.0000000000, 222.0000000000, 164.0000000000, 227.↩
0000000000, 165.0000000000, 159.0000000000, 165.0000000000, 182.0000000000, 176.0000000000,
166.0000000000, 177.0000000000, 183.0000000000, 178.0000000000, 177.0000000000, 166.↩
0000000000, 167.0000000000, 178.0000000000, 179.0000000000, 167.0000000000, 168.0000000000,
161.0000000000, 184.0000000000, 179.0000000000, 180.0000000000, 168.0000000000, 175.↩
0000000000, 169.0000000000, 185.0000000000, 180.0000000000, 186.0000000000, 174.0000000000,
169.0000000000, 181.0000000000, 70.0000000000, 63.0000000000, 227.0000000000, 165.0000000000,
182.0000000000, 170.0000000000, 171.0000000000, 171.0000000000, 183.0000000000, 182.↩
0000000000, 166.0000000000, 172.0000000000, 188.0000000000, 183.0000000000, 184.0000000000,
172.0000000000, 167.0000000000, 173.0000000000, 168.0000000000, 190.0000000000, 174.↩
0000000000, 184.0000000000, 173.0000000000, 185.0000000000, 175.0000000000, 186.0000000000,
174.0000000000, 185.0000000000, 175.0000000000, 192.0000000000, 186.0000000000, 70.0000000000,
187.0000000000, 171.0000000000, 176.0000000000, 227.0000000000, 237.0000000000, 232.↩
0000000000, 177.0000000000, 188.0000000000, 189.0000000000, 184.0000000000, 188.0000000000,
172.0000000000, 178.0000000000, 177.0000000000, 189.0000000000, 178.0000000000, 179.↩
0000000000, 190.0000000000, 183.0000000000, 173.0000000000, 174.0000000000, 190.0000000000,
179.0000000000, 186.0000000000, 180.0000000000, 191.0000000000, 191.0000000000, 175.↩
0000000000, 192.0000000000, 185.0000000000, 180.0000000000, 181.0000000000, 193.0000000000,
70.0000000000, 74.0000000000, 192.0000000000, 181.0000000000, 189.0000000000, 194.0000000000,
182.0000000000, 237.0000000000, 177.0000000000, 183.0000000000, 242.0000000000, 188.↩
0000000000, 190.0000000000, 195.0000000000, 184.0000000000, 183.0000000000, 194.0000000000,
189.0000000000, 185.0000000000, 200.0000000000, 196.0000000000, 179.0000000000, 195.↩
0000000000, 184.0000000000, 191.0000000000, 196.0000000000, 192.0000000000, 201.0000000000,
197.0000000000, 185.0000000000, 186.0000000000, 190.0000000000, 186.0000000000, 181.↩
0000000000, 193.0000000000, 198.0000000000, 197.0000000000, 191.0000000000, 187.0000000000,
198.0000000000, 187.0000000000, 74.0000000000, 192.0000000000, 199.0000000000, 189.0000000000,
188.0000000000, 242.0000000000, 195.0000000000, 199.0000000000, 200.0000000000, 203.↩
0000000000, 190.0000000000, 194.0000000000, 189.0000000000, 191.0000000000, 190.0000000000,
201.0000000000, 200.0000000000, 198.0000000000, 192.0000000000, 205.0000000000, 201.↩
0000000000, 191.0000000000, 202.0000000000, 202.0000000000, 192.0000000000, 197.0000000000,
242.0000000000, 247.0000000000, 203.0000000000, 252.0000000000, 195.0000000000, 194.↩
0000000000, 195.0000000000, 190.0000000000, 203.0000000000, 204.0000000000, 201.0000000000,
196.0000000000, 207.0000000000, 196.0000000000, 197.0000000000, 191.0000000000, 200.↩
0000000000, 204.0000000000, 205.0000000000, 197.0000000000, 205.0000000000, 206.0000000000,
204.0000000000, 195.0000000000, 200.0000000000, 252.0000000000, 257.0000000000, 199.↩
0000000000, 203.0000000000, 201.0000000000, 200.0000000000, 207.0000000000, 206.0000000000,
207.0000000000, 197.0000000000, 201.0000000000, 204.0000000000, 257.0000000000, 203.↩
0000000000, 207.0000000000, 208.0000000000, 261.0000000000, 206.0000000000, 201.0000000000,
208.0000000000, 204.0000000000, 206.0000000000, 261.0000000000, 211.0000000000, 97.0000000000,
210.0000000000, 211.0000000000, 97.0000000000, 212.0000000000, 213.0000000000, 218.0000000000,
209.0000000000, 216.0000000000, 210.0000000000, 217.0000000000, 212.0000000000, 223.↩
0000000000, 210.0000000000, 217.0000000000, 211.0000000000, 218.0000000000, 224.0000000000,
104.0000000000, 218.0000000000, 210.0000000000, 219.0000000000, 97.0000000000, 163.↩
0000000000, 164.0000000000, 221.0000000000, 215.0000000000, 164.0000000000, 222.0000000000,
214.0000000000, 221.0000000000, 170.0000000000, 217.0000000000, 220.0000000000, 223.↩

0000000000, 221.0000000000, 212.0000000000, 220.0000000000, 216.0000000000, 211.0000000000, 223.0000000000, 224.0000000000, 213.0000000000, 212.0000000000, 210.0000000000, 228.↩
0000000000, 224.0000000000, 225.0000000000, 104.0000000000, 110.0000000000, 229.0000000000, 213.0000000000, 214.0000000000, 221.0000000000, 217.0000000000, 214.0000000000, 223.↩
0000000000, 215.0000000000, 226.0000000000, 217.0000000000, 220.0000000000, 222.0000000000, 170.0000000000, 215.0000000000, 226.0000000000, 227.0000000000, 221.0000000000, 221.↩
0000000000, 228.0000000000, 217.0000000000, 218.0000000000, 212.0000000000, 226.0000000000, 233.0000000000, 228.0000000000, 213.0000000000, 229.0000000000, 219.0000000000, 218.↩
0000000000, 219.0000000000, 229.0000000000, 110.0000000000, 116.0000000000, 230.0000000000, 234.0000000000, 223.0000000000, 228.0000000000, 222.0000000000, 221.0000000000, 231.↩
0000000000, 227.0000000000, 231.0000000000, 232.0000000000, 176.0000000000, 226.0000000000, 222.0000000000, 170.0000000000, 182.0000000000, 224.0000000000, 233.0000000000, 231.↩
0000000000, 226.0000000000, 218.0000000000, 223.0000000000, 236.0000000000, 233.0000000000, 234.0000000000, 219.0000000000, 225.0000000000, 224.0000000000, 235.0000000000, 122.↩
0000000000, 225.0000000000, 234.0000000000, 116.0000000000, 226.0000000000, 236.0000000000, 228.0000000000, 232.0000000000, 227.0000000000, 231.0000000000, 227.0000000000, 182.↩
0000000000, 236.0000000000, 237.0000000000, 234.0000000000, 236.0000000000, 224.0000000000, 238.0000000000, 229.0000000000, 241.0000000000, 228.0000000000, 225.0000000000, 238.↩
0000000000, 230.0000000000, 243.0000000000, 229.0000000000, 235.0000000000, 233.0000000000, 239.0000000000, 239.0000000000, 240.0000000000, 128.0000000000, 122.0000000000, 234.↩
0000000000, 230.0000000000, 231.0000000000, 228.0000000000, 232.0000000000, 237.0000000000, 241.0000000000, 233.0000000000, 182.0000000000, 242.0000000000, 236.0000000000, 232.↩
0000000000, 188.0000000000, 241.0000000000, 243.0000000000, 246.0000000000, 233.0000000000, 234.0000000000, 241.0000000000, 244.0000000000, 248.0000000000, 243.0000000000, 234.↩
0000000000, 235.0000000000, 240.0000000000, 137.0000000000, 249.0000000000, 128.0000000000, 235.0000000000, 133.0000000000, 239.0000000000, 245.0000000000, 244.0000000000, 242.↩
0000000000, 233.0000000000, 236.0000000000, 237.0000000000, 238.0000000000, 246.0000000000, 241.0000000000, 194.0000000000, 199.0000000000, 247.0000000000, 237.0000000000, 251.↩
0000000000, 188.0000000000, 246.0000000000, 234.0000000000, 248.0000000000, 239.0000000000, 251.0000000000, 238.0000000000, 246.0000000000, 248.0000000000, 249.0000000000, 253.↩
0000000000, 240.0000000000, 239.0000000000, 249.0000000000, 137.0000000000, 254.0000000000, 240.0000000000, 250.0000000000, 241.0000000000, 238.0000000000, 251.0000000000, 242.↩
0000000000, 243.0000000000, 251.0000000000, 199.0000000000, 252.0000000000, 242.0000000000, 251.0000000000, 243.0000000000, 239.0000000000, 244.0000000000, 253.0000000000, 244.↩
0000000000, 254.0000000000, 258.0000000000, 253.0000000000, 240.0000000000, 245.0000000000, 137.0000000000, 140.0000000000, 254.0000000000, 245.0000000000, 255.0000000000, 259.↩
0000000000, 253.0000000000, 243.0000000000, 252.0000000000, 256.0000000000, 242.0000000000, 247.0000000000, 248.0000000000, 246.0000000000, 251.0000000000, 203.0000000000, 199.↩
0000000000, 247.0000000000, 256.0000000000, 257.0000000000, 260.0000000000, 260.0000000000, 256.0000000000, 258.0000000000, 244.0000000000, 248.0000000000, 249.0000000000, 251.↩
0000000000, 258.0000000000, 245.0000000000, 259.0000000000, 249.0000000000, 250.0000000000, 140.0000000000, 142.0000000000, 250.0000000000, 260.0000000000, 253.0000000000, 251.↩
0000000000, 252.0000000000, 252.0000000000, 203.0000000000, 263.0000000000, 261.0000000000, 206.0000000000, 260.0000000000, 263.0000000000, 262.0000000000, 249.0000000000, 260.↩
0000000000, 254.0000000000, 253.0000000000, 259.0000000000, 250.0000000000, 255.0000000000, 258.0000000000, 254.0000000000, 256.0000000000, 252.0000000000, 253.0000000000, 258.↩
0000000000, 263.0000000000, 257.0000000000, 263.0000000000, 206.0000000000, 257.0000000000, 258.0000000000, 259.0000000000, 262.0000000000, 260.0000000000, 258.0000000000, 257.↩
0000000000],[0.0000000000, 1.0000000000, 2.0000000000, 3.0000000000, 4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000, 9.0000000000, 10.0000000000, 11.0000000000, 12.↩
0000000000, 13.0000000000, 14.0000000000, 15.0000000000, 16.0000000000, 17.0000000000, 18.↩
0000000000, 19.0000000000, 20.0000000000, 21.0000000000, 22.0000000000, 23.0000000000, 24.↩
0000000000, 25.0000000000, 26.0000000000, 27.0000000000, 28.0000000000, 29.0000000000, 30.↩
0000000000, 31.0000000000, 32.0000000000, 33.0000000000, 34.0000000000, 35.0000000000, 36.↩
0000000000, 37.0000000000, 38.0000000000, 39.0000000000, 40.0000000000, 41.0000000000, 42.↩
0000000000, 43.0000000000, 44.0000000000, 45.0000000000, 46.0000000000, 47.0000000000, 48.↩
0000000000, 49.0000000000, 50.0000000000, 51.0000000000, 52.0000000000, 53.0000000000, 54.↩
0000000000, 55.0000000000, 56.0000000000, 57.0000000000, 58.0000000000, 59.0000000000, 60.↩

0000000000, 61.0000000000, 62.0000000000, 63.0000000000, 64.0000000000, 65.0000000000, 66.↩
0000000000, 67.0000000000, 68.0000000000, 69.0000000000, 70.0000000000, 71.0000000000, 72.↩
0000000000, 73.0000000000, 74.0000000000, 75.0000000000, 76.0000000000, 77.0000000000, 78.↩
0000000000, 79.0000000000, 80.0000000000, 81.0000000000, 82.0000000000, 83.0000000000, 84.↩
0000000000, 85.0000000000, 86.0000000000, 87.0000000000, 88.0000000000, 89.0000000000, 90.↩
0000000000, 91.0000000000, 92.0000000000, 93.0000000000, 94.0000000000, 95.0000000000, 96.↩
0000000000, 97.0000000000, 98.0000000000, 99.0000000000, 100.0000000000, 101.0000000000, 102.↩
0000000000, 103.0000000000, 104.0000000000, 105.0000000000, 106.0000000000, 107.0000000000,
108.0000000000, 109.0000000000, 110.0000000000, 111.0000000000, 112.0000000000, 113.↩
0000000000, 114.0000000000, 115.0000000000, 116.0000000000, 117.0000000000, 118.0000000000,
119.0000000000, 120.0000000000, 121.0000000000, 122.0000000000, 123.0000000000, 124.↩
0000000000, 125.0000000000, 126.0000000000, 127.0000000000, 128.0000000000, 129.0000000000,
130.0000000000, 131.0000000000, 132.0000000000, 133.0000000000, 134.0000000000, 135.↩
0000000000, 136.0000000000, 137.0000000000, 138.0000000000, 139.0000000000, 140.0000000000,
141.0000000000, 142.0000000000, 143.0000000000, 144.0000000000, 145.0000000000, 146.↩
0000000000, 147.0000000000, 148.0000000000, 149.0000000000, 150.0000000000, 151.0000000000,
152.0000000000, 153.0000000000, 154.0000000000, 155.0000000000, 156.0000000000, 157.↩
0000000000, 158.0000000000, 159.0000000000, 160.0000000000, 161.0000000000, 162.0000000000,
163.0000000000, 164.0000000000, 165.0000000000, 166.0000000000, 167.0000000000, 168.↩
0000000000, 169.0000000000, 170.0000000000, 171.0000000000, 172.0000000000, 173.0000000000,
174.0000000000, 175.0000000000, 176.0000000000, 177.0000000000, 178.0000000000, 179.↩
0000000000, 180.0000000000, 181.0000000000, 182.0000000000, 183.0000000000, 184.0000000000,
185.0000000000, 186.0000000000, 187.0000000000, 188.0000000000, 189.0000000000, 190.↩
0000000000, 191.0000000000, 192.0000000000, 193.0000000000, 194.0000000000, 195.0000000000,
196.0000000000, 197.0000000000, 198.0000000000, 199.0000000000, 200.0000000000, 201.↩
0000000000, 202.0000000000, 203.0000000000, 204.0000000000, 205.0000000000, 206.0000000000,
207.0000000000, 208.0000000000, 209.0000000000, 210.0000000000, 211.0000000000, 212.↩
0000000000, 213.0000000000, 214.0000000000, 215.0000000000, 216.0000000000, 217.0000000000,
218.0000000000, 219.0000000000, 220.0000000000, 221.0000000000, 222.0000000000, 223.↩
0000000000, 224.0000000000, 225.0000000000, 226.0000000000, 227.0000000000, 228.0000000000,
229.0000000000, 230.0000000000, 231.0000000000, 232.0000000000, 233.0000000000, 234.↩
0000000000, 235.0000000000, 236.0000000000, 237.0000000000, 238.0000000000, 239.0000000000,
240.0000000000, 241.0000000000, 242.0000000000, 243.0000000000, 244.0000000000, 245.↩
0000000000, 246.0000000000, 247.0000000000, 248.0000000000, 249.0000000000, 250.0000000000,
251.0000000000, 252.0000000000, 253.0000000000, 254.0000000000, 255.0000000000, 256.↩
0000000000, 257.0000000000, 258.0000000000, 259.0000000000, 260.0000000000, 261.0000000000,
262.0000000000, 263.0000000000, 0.0000000000, 0.0000000000, 1.0000000000, 1.0000000000, 1.↩
0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 2.0000000000, 2.0000000000,
2.0000000000, 2.0000000000, 3.0000000000, 3.0000000000, 3.0000000000, 3.0000000000, 3.↩
0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000,
5.0000000000, 5.0000000000, 5.0000000000, 6.0000000000, 6.0000000000, 6.0000000000, 6.↩
0000000000, 6.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 8.0000000000,
8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 9.↩
0000000000, 9.0000000000, 9.0000000000, 9.0000000000, 9.0000000000, 10.0000000000, 10.↩
0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 11.↩
0000000000, 11.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.↩
0000000000, 12.0000000000, 13.0000000000, 13.0000000000, 13.0000000000, 14.0000000000, 14.↩
0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 15.0000000000, 15.↩
0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 16.0000000000, 16.↩
0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 17.0000000000, 17.↩
0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 18.↩
0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 18.↩
0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 19.↩
0000000000, 20.0000000000, 20.0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 21.↩
0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 22.0000000000, 22.0000000000, 22.↩
0000000000, 22.0000000000, 22.0000000000, 22.0000000000, 23.0000000000, 23.0000000000, 23.↩
0000000000, 23.0000000000, 23.0000000000, 24.0000000000, 24.0000000000, 24.0000000000, 24.↩
0000000000, 24.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.↩

0000000000, 25.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 26.↩
0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 27.↩
0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 28.↩
0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 29.↩
0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.↩
0000000000, 30.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.↩
0000000000, 31.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.↩
0000000000, 32.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.↩
0000000000, 33.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.↩
0000000000, 34.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.↩
0000000000, 35.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 36.↩
0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 37.↩
0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 38.↩
0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 39.↩
0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 40.↩
0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 41.↩
0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.↩
0000000000, 42.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.↩
0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 44.0000000000, 44.0000000000, 44.↩
0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 45.0000000000, 45.↩
0000000000, 45.0000000000, 45.0000000000, 45.0000000000, 46.0000000000, 46.0000000000, 46.↩
0000000000, 46.0000000000, 46.0000000000, 46.0000000000, 47.0000000000, 47.0000000000, 47.↩
0000000000, 47.0000000000, 47.0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 48.↩
0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 49.0000000000, 49.0000000000, 49.↩
0000000000, 49.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 50.↩
0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 51.0000000000, 52.↩
0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.↩
0000000000, 52.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 53.↩
0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.↩
0000000000, 54.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 55.↩
0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 56.↩
0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 57.↩
0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 58.0000000000, 59.↩
0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 59.0000000000, 60.↩
0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 60.0000000000, 61.↩
0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 61.↩
0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 62.↩
0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.↩
0000000000, 63.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.↩
0000000000, 64.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.↩
0000000000, 65.0000000000, 65.0000000000, 66.0000000000, 66.0000000000, 67.0000000000, 67.↩
0000000000, 67.0000000000, 67.0000000000, 68.0000000000, 68.0000000000, 68.0000000000, 68.↩
0000000000, 68.0000000000, 68.0000000000, 69.0000000000, 69.0000000000, 69.0000000000, 70.↩
0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 70.↩
0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 71.0000000000, 72.↩
0000000000, 72.0000000000, 72.0000000000, 73.0000000000, 73.0000000000, 74.0000000000, 74.↩
0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 75.0000000000, 75.↩
0000000000, 75.0000000000, 75.0000000000, 76.0000000000, 76.0000000000, 77.0000000000, 77.↩
0000000000, 77.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.↩
0000000000, 78.0000000000, 79.0000000000, 79.0000000000, 79.0000000000, 79.0000000000, 80.↩
0000000000, 80.0000000000, 80.0000000000, 80.0000000000, 81.0000000000, 81.0000000000, 81.↩
0000000000, 81.0000000000, 81.0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 82.↩
0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 83.0000000000, 83.0000000000, 84.↩
0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 84.↩
0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 85.↩
0000000000, 86.0000000000, 86.0000000000, 86.0000000000, 86.0000000000, 87.0000000000, 87.↩
0000000000, 87.0000000000, 87.0000000000, 87.0000000000, 88.0000000000, 88.0000000000, 88.↩
0000000000, 88.0000000000, 88.0000000000, 88.0000000000, 89.0000000000, 89.0000000000, 89.↩

0000000000, 89.0000000000, 89.0000000000, 89.0000000000, 90.0000000000, 90.0000000000, 90.↩
0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 91.↩
0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 91.0000000000, 92.↩
0000000000, 92.0000000000, 92.0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 93.↩
0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 94.0000000000, 94.0000000000, 94.↩
0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 95.0000000000, 95.↩
0000000000, 95.0000000000, 95.0000000000, 95.0000000000, 96.0000000000, 96.0000000000, 96.↩
0000000000, 96.0000000000, 96.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.↩
0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 98.0000000000, 98.↩
0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 99.0000000000, 100.↩
0000000000, 100.0000000000, 100.0000000000, 100.0000000000, 101.0000000000, 101.0000000000,
101.0000000000, 101.0000000000, 101.0000000000, 101.0000000000, 102.0000000000, 102.↩
0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000,
103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.↩
0000000000, 103.0000000000, 104.0000000000, 104.0000000000, 104.0000000000, 104.0000000000,
104.0000000000, 104.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.↩
0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 106.0000000000,
106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.↩
0000000000, 107.0000000000, 107.0000000000, 107.0000000000, 107.0000000000, 107.0000000000,
108.0000000000, 108.0000000000, 108.0000000000, 108.0000000000, 108.0000000000, 109.↩
0000000000, 109.0000000000, 109.0000000000, 109.0000000000, 109.0000000000, 110.0000000000,
110.0000000000, 110.0000000000, 110.0000000000, 110.0000000000, 111.0000000000, 111.↩
0000000000, 111.0000000000, 111.0000000000, 112.0000000000, 112.0000000000, 112.0000000000,
112.0000000000, 112.0000000000, 112.0000000000, 113.0000000000, 113.0000000000, 113.↩
0000000000, 113.0000000000, 113.0000000000, 113.0000000000, 114.0000000000, 114.0000000000,
114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 115.0000000000, 115.↩
0000000000, 115.0000000000, 115.0000000000, 115.0000000000, 116.0000000000, 116.0000000000,
116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 117.↩
0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000,
118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.↩
0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000,
119.0000000000, 119.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.↩
0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 121.0000000000,
121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 121.↩
0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000,
122.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.↩
0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 124.0000000000, 124.0000000000,
124.0000000000, 124.0000000000, 124.0000000000, 125.0000000000, 125.0000000000, 125.↩
0000000000, 125.0000000000, 126.0000000000, 126.0000000000, 126.0000000000, 126.0000000000,
127.0000000000, 127.0000000000, 127.0000000000, 128.0000000000, 128.0000000000, 128.↩
0000000000, 128.0000000000, 128.0000000000, 129.0000000000, 129.0000000000, 129.0000000000,
129.0000000000, 129.0000000000, 129.0000000000, 130.0000000000, 130.0000000000, 130.↩
0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 131.0000000000,
131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.↩
0000000000, 131.0000000000, 132.0000000000, 132.0000000000, 132.0000000000, 132.0000000000,
133.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 133.0000000000, 134.↩
0000000000, 134.0000000000, 134.0000000000, 134.0000000000, 134.0000000000, 135.0000000000,
135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 136.↩
0000000000, 136.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000,
137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 138.0000000000, 138.↩
0000000000, 138.0000000000, 138.0000000000, 138.0000000000, 139.0000000000, 139.0000000000,
139.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 140.↩
0000000000, 140.0000000000, 141.0000000000, 141.0000000000, 141.0000000000, 141.0000000000,
142.0000000000, 142.0000000000, 143.0000000000, 143.0000000000, 143.0000000000, 143.↩
0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 144.0000000000,
145.0000000000, 145.0000000000, 145.0000000000, 146.0000000000, 146.0000000000, 146.↩
0000000000, 146.0000000000, 146.0000000000, 146.0000000000, 147.0000000000, 147.0000000000,
147.0000000000, 147.0000000000, 147.0000000000, 148.0000000000, 148.0000000000, 149.↩

0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.↩
0000000000, 150.0000000000, 150.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 152.0000000000, 152.↩
0000000000, 152.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 153.0000000000, 154.0000000000, 154.0000000000, 154.↩
0000000000, 154.0000000000, 154.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 155.0000000000, 156.0000000000, 156.0000000000, 156.0000000000, 156.↩
0000000000, 156.0000000000, 156.0000000000, 157.0000000000, 157.0000000000, 157.0000000000, 157.0000000000, 158.0000000000, 158.0000000000, 158.0000000000, 158.0000000000, 158.↩
0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 160.↩
0000000000, 160.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 162.0000000000, 162.0000000000, 162.0000000000, 162.↩
0000000000, 162.0000000000, 162.0000000000, 163.0000000000, 163.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.↩
0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.↩
0000000000, 166.0000000000, 166.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 167.0000000000, 168.0000000000, 168.0000000000, 168.↩
0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 170.↩
0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 171.↩
0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 172.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.0000000000, 173.↩
0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 174.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 175.↩
0000000000, 175.0000000000, 175.0000000000, 176.0000000000, 176.0000000000, 176.0000000000, 176.0000000000, 176.0000000000, 177.0000000000, 177.0000000000, 177.0000000000, 177.↩
0000000000, 177.0000000000, 177.0000000000, 178.0000000000, 178.0000000000, 178.0000000000, 178.0000000000, 178.0000000000, 179.0000000000, 179.0000000000, 179.0000000000, 179.↩
0000000000, 179.0000000000, 179.0000000000, 180.0000000000, 180.0000000000, 180.0000000000, 180.0000000000, 181.0000000000, 181.0000000000, 181.0000000000, 181.0000000000, 181.↩
0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 183.0000000000, 183.0000000000, 183.↩
0000000000, 183.0000000000, 183.0000000000, 184.0000000000, 184.0000000000, 184.0000000000, 184.0000000000, 184.0000000000, 184.0000000000, 185.0000000000, 185.0000000000, 185.↩
0000000000, 185.0000000000, 185.0000000000, 185.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 187.0000000000, 187.↩
0000000000, 187.0000000000, 187.0000000000, 187.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 189.↩
0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.↩
0000000000, 190.0000000000, 190.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 192.0000000000, 192.↩
0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 193.0000000000, 193.0000000000, 193.0000000000, 193.0000000000, 194.0000000000, 194.↩
0000000000, 194.0000000000, 194.0000000000, 194.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 196.0000000000, 196.↩
0000000000, 196.0000000000, 196.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 198.0000000000, 198.0000000000, 198.↩
0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 200.↩
0000000000, 200.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 202.0000000000, 202.0000000000, 203.↩
0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 204.0000000000, 204.0000000000, 204.0000000000, 204.0000000000, 204.↩

0000000000, 205.0000000000, 205.0000000000, 205.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 207.0000000000, 207.↩
0000000000, 207.0000000000, 207.0000000000, 208.0000000000, 208.0000000000, 209.0000000000, 209.0000000000, 209.0000000000, 210.0000000000, 210.0000000000, 210.0000000000, 210.↩
0000000000, 210.0000000000, 210.0000000000, 211.0000000000, 211.0000000000, 211.0000000000, 211.0000000000, 212.0000000000, 212.0000000000, 212.0000000000, 212.0000000000, 212.↩
0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 214.0000000000, 214.0000000000, 214.0000000000, 214.0000000000, 215.↩
0000000000, 215.0000000000, 215.0000000000, 215.0000000000, 215.0000000000, 216.0000000000, 216.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 217.↩
0000000000, 217.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 219.0000000000, 219.0000000000, 219.0000000000, 219.↩
0000000000, 219.0000000000, 219.0000000000, 220.0000000000, 220.0000000000, 220.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.↩
0000000000, 221.0000000000, 222.0000000000, 222.0000000000, 222.0000000000, 222.0000000000, 222.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 223.↩
0000000000, 223.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 225.0000000000, 225.0000000000, 225.0000000000, 225.↩
0000000000, 225.0000000000, 225.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 227.0000000000, 227.0000000000, 227.↩
0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.↩
0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 231.↩
0000000000, 231.0000000000, 231.0000000000, 231.0000000000, 231.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 232.0000000000, 233.0000000000, 233.↩
0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.↩
0000000000, 234.0000000000, 234.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 236.0000000000, 236.0000000000, 236.↩
0000000000, 236.0000000000, 236.0000000000, 236.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 238.0000000000, 238.↩
0000000000, 238.0000000000, 238.0000000000, 238.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 240.0000000000, 240.↩
0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 241.0000000000, 241.0000000000, 241.0000000000, 241.0000000000, 241.↩
0000000000, 241.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 243.0000000000, 243.↩
0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 245.0000000000, 245.↩
0000000000, 245.0000000000, 245.0000000000, 245.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 247.0000000000, 247.↩
0000000000, 247.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 248.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 249.↩
0000000000, 249.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 251.↩
0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.↩
0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 254.0000000000, 254.0000000000, 254.↩
0000000000, 254.0000000000, 255.0000000000, 255.0000000000, 255.0000000000, 256.0000000000, 256.0000000000, 256.0000000000, 256.0000000000, 257.0000000000, 257.0000000000, 257.↩
0000000000, 257.0000000000, 257.0000000000, 257.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 259.↩
0000000000, 259.0000000000, 259.0000000000, 259.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 261.0000000000, 261.↩
0000000000, 261.0000000000, 262.0000000000, 262.0000000000, 263.0000000000, 263.0000000000, 263.0000000000, 263.0000000000, 263.0000000000]

- list testCase.calcWeightsGaussCorrect = [[-0.57735026918962573105886800, 0.57735026918962573105886800],[1.↩
000000000000004440892099, 1.000000000000004440892099]]
- list testCase.gaußLegendreCorrect = [[0.21132486540518713, 0.21132486540518713, 0.7886751345948129, 0.7886751345948129], [0.16666666666666669, 0.6220084679281462, 0.044658198738520456, 0.↩
16666666666666669], [0.19716878364870338, 0.19716878364870338, 0.052831216351296825, 0.↩
052831216351296825]]
- list testCase.reducedSFCorrect = [-0.27937980331459932870075136, -0.33966732733873783445233130, 158.30535989173870348167838529, 111.95922396290268352458951995, 58.10903859010657157568857656, 193.39790162602255918500304688, 268.25033740186444219943950884, 189.82215394456304125014867168, 0.25883372957144956671982072, 98.74870633402809971812530421, 215.53653410699894266144838184, 327.39158447388479089568136260, 284.96744613445838467669091187, 202.31101004349304162133194041, 0.82513685120877511280923500, 105.22484885820357192187657347, 223.08256339289710012963041663, 364.76793579171919645887101069, 347.61723438903078431394533254, 186.13386039326010745753592346, 132.63331666423582078095932957, 0.93045442511324694123686641, 69.59377646537200234888587147, 377.49981102452846926098573022, 387.86131916037487599169253372, 228.06934028761415333974582609, -0.39282069172599420880942489, 0.28340911989108974466944346, 1.32922047073893168089853134, 0.77036064878831922442259383, 402.01980094098689733073115349, 253.64580857732585172925610095, -0.71137220262248490598722128, -187.02281867595976905249699485, -132.12396908339931655973487068, 1.00935338967148879874002887, -68.31938968731270733769633807, 262.84597566425361492292722687, -1.01378116441184573659484158, -229.49291759567395843077974860, -285.00916079306904293844127096, -202.15395986486728929776290897, 0.20369749822588545384860481, -105.12807714412967641237628413, -1.65128343645850605980740511, -255.65451390973146317264763638, -348.80806087323372821629163809, -268.84160316836425863584736362, -190.14226619109740568092092872, -0.35372141836276682624884415, -98.67697413471481127089646179, -265.72733211685635978938080370, -390.53074044891235416798735969, -328.54385597348704095566063188, -159.10063256399396891538344789, -112.66830824900223717577318894, -0.48136222438574449711268244, -58.49296577544600950204767287, -406.26733909792579879649565555, -367.00125201198864033358404413, -194.68519835904922388181148563, -0.38739421188725309264100360, -380.72973951162691719218855724, -216.95201141250419141215388663, -224.64681969190800714386568870, -57.64885629013803480802380363, -111.76778906363414023417135468, -97.23151838524054824119957630, -158.57787734712869109898747411, -189.13083157351150020986096933, -103.47360646666479055966192391, -194.37812613590944010866223834, -268.14533020187656120469910093, -201.00144759920507908645959105, -67.33160591182809184829238802, -216.97908089605186887638410553, -328.52149241710583282838342711, -284.82532200213120177068049088, -130.61038743255170402335352264, 1.17408313059079461027067737, -225.01886593214194931533711497, -366.28282561516010673585697077, -347.97810273027931771139265038, -184.84955361689296182703401428, 1.35575884552714853725774447, 68.83865863464234280399978161, -379.59834765834438030651654117, -389.38131238244437781759188510, -226.65542721376317558679147623, 2.45142690491787362816467066, 132.61870525252811603422742337, 104.51756439557146904917317443, -402.65102785189810674637556076, -252.04129957093991265537624713, 2.66640808738427992352626461, 188.46031263508777442439168226, 202.68330394428298291131795850, 98.04919160612566031431924785, -262.11224472208607494394527748, 2.80043348997752872264754842, 231.30365585971119912755966652, 286.80511684477886547028901987, 190.23232474250315249264531303, 57.84805143771897917304158909, 2.85708460601647828980276245, 258.69203442119561486833845265, 350.58867061999404768357635476, 268.52259877181154479330871254, 112.10337467749660333993233507, 266.08331708857917874411214143, 390.90976985716883973509538919, 329.01088679331462572008604184, 158.71128445003517981604090892, 404.59245903407168043486308306, 366.35434029559155533206649125, 194.19573855113847571374208201, 379.94898461640434561559231952, 216.17590064203761812677839771, 223.85736931998752652361872606, 215.30222433132459514126821887, 193.26277766072317376710998360, 364.27701325326768255763454363, 157.72987052894848147843731567, 326.94128157610799689791747369, 388.09280021282381767377955839, 111.18402577739000491874321597, 266.54277820804679777211276814, 346.73233678344666941484319977, 252.57711524406573744272463955, 57.01639165715037194104297669, 187.85218553479762704228051007, 282.67168803322431358537869528, 224.67611712193377115909243003, -3.14326924385573391873549554, -0.90010570311670168308637585, 96.17069437543644028210110264, 199.33582061541360985756909940, 182.27846100397448481089668348, -4.57611186272201919678082049, -257.99057047583272606061655097, -1.82557398572480522602745623, 101.69189406672646214246924501, 127.54445573081555664884945145, -4.49732115287934419711746159, -232.30379071310053973320464138, -392.35336998126877006143331528, -2.65962064013652721428115910, 64.23306507635152229340746999, -4.17823235146674676343536703, -190.62136057342786443769000471, -350.97981405658060793939512223, -367.33195935840376478154212236, -3.92525467655692494872710085, -4.21737374407493348371644970, -136.02000312436450712993973866,

-288.23032697439572302755550481, -329.76997539886201593617442995, -217.04860831373792962040170096, -4.22506971989652058141473390, -72.29251053978725849447073415, -205.62211432499287866448867135, -270.03888658781465892388951033, -195.10425396920604157458001282, -3.75470502920668458557429403, -107.71214521643315720211830921, -191.64672875892048864443495404, -159.55913514871488700919144321, -2.82958247867568246647351771, -100.33839131188909732372849248, -113.28188975877471023068210343, -2.02984471265764909730933141, -59.20137760431371987124293810, -1.13006153433920530915202107, -216.94986764694439784761925694, -194.17815277072924118328955956, -366.46414265432213142048567533, -58.68817436396830089506693184, -158.58371851794368012633640319, -328.49641010394003615147084929, -389.37906114078765540398308076, -112.59894550018677250591281336, -99.47273320793448192489449866, -268.50544005644428580126259476, -349.78532743252520731402910315, -254.30643984757679731956159230, -190.57684701519576719874748960, -106.72241271551203567469201516, -286.77766766087427185993874446, -228.97129601689493938465602696, 0.65888959003758174048925866, -203.43166351223138121895317454, -71.95880175563581815367797390, -188.94812353702479867934016511, -0.54028097490086324228286685, 255.71015351091924117099551950, -135.09478864364308492440613918, -4.18373247132461756336851977, -1.97138030481194448384485440, 228.04006865312763352449110243, 391.09379142082968883187277243, -3.47586424690169337736733723, 65.01418211509854927498963661, 184.65606105019628557784017175, 348.36366144437067760009085760, 366.20261451786950601672288030, 128.93974257358527779615542386, 101.83529783301392512839811388, 283.51551025120562599113327451, 327.60501736958264018539921381, 216.03661549146309539537469391, 200.00439085138680184172699228, 96.43013782801264710542454850, 267.22322299296490655251545832, 193.49255409229277802296564914, 188.08418865096339800402347464, 57.04436608707574407617357792, 157.96464661316298361271037720, 111.31922987220724508006242104]

- list testCase.SFCorrectValue = [-0.27937980331459933, 158.30535989173870348, -0.27937980331459933, 111.95922396290268352, 58.10903859010657158, -0.27937980331459933, 193.39790162602255919, -0.27937980331459933, 268.25033740186444220, 189.82215394456304125, 0.25883372957144957, -0.27937980331459933, 98.74870633402809972, -0.27937980331459933, 215.53653410699894266, 327.39158447388479090, 284.96744613445838468, 202.31101004349304162, 0.82513685120877511, 105.22484885820357192, -0.27937980331459933, 223.08256339289710013, 364.76793579171919646, 347.61723438903078431, 186.13386039326010746, 132.63331666423582078, 0.93045442511324694, 69.59377646537200235, 377.49981102452846926, 387.86131916037487599, 228.06934028761415334, -0.39282069172599421, 0.28340911989108974, 1.32922047073893168, 0.77036064878831922, 402.↩
01980094098689733, 253.64580857732585173, -0.71137220262248491, -187.02281867595976905, -132.12396908339931656, 1.00935338967148880, -68.31938968731270734, 262.84597566425361492, -1.01378116441184574, -229.49291759567395843, -285.00916079306904294, -202.15395986486728930, 0.20369749822588545, -105.12807714412967641, -1.65128343645850606, -255.65451390973146317, -348.80806087323372822, -268.84160316836425864, -190.14226619109740568, -0.35372141836276683, -98.67697413471481127, -265.72733211685635979, -390.53074044891235417, -328.54385597348704096, -159.10063256399396892, -112.66830824900223718, -0.48136222438574450, -58.49296577544600950, -406.26733909792579880, -367.00125201198864033, -194.68519835904922388, -0.33966732733873783, -0.33966732733873783, -0.38739421188725309, -0.33966732733873783, -380.72973951162691719, -216.95201141250419141, -0.33966732733873783, -0.33966732733873783, -224.64681969190800714, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, -57.64885629013803481, -0.↩
27937980331459933, -111.76778906363414023, -97.23151838524054824, -158.57787734712869110, -0.27937980331459933, -189.13083157351150021, -103.47360646666479056, -0.27937980331459933, -194.37812613590944011, -268.14533020187656120, -201.00144759920507909, -67.33160591182809185, -216.97908089605186888, -0.27937980331459933, -328.52149241710583283, -284.82532200213120177, -130.61038743255170402, 1.17408313059079461, -225.01886593214194932, -0.27937980331459933, -366.28282561516010674, -347.97810273027931771, -184.84955361689296183, 1.35575884552714854, 68.83865863464234280, -379.59834765834438031, -389.38131238244437782, -226.65542721376317559, 2.45142690491787363, 132.61870525252811603, 104.51756439557146905, -402.65102785189810675, -252.04129957093991266, 2.66640808738427992, 188.46031263508777442, 202.68330394428298291, 98.04919160612566031, -262.11224472208607494, 2.80043348997752872, 231.30365585971119913, 286.80511684477886547, 190.23232474250315249, 57.84805143771897917, 2.85708460601647829, 258.69203442119561487, 350.58867061999404768, 268.52259877181154479, 112.10337467749660334, -0.33966732733873783, 266.08331708857917874, 390.90976985716883974, 329.01088679331462572, 158.71128445003517982, -0.33966732733873783, 404.59245903407168043, 366.35434029559155533, 194.19573855113847571, -0.33966732733873783, 379.94898461640434562, 216.17590064203761813, -0.33966732733873783, 223.85736931998752652, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, 215.30222433132459514, -0.27937980331459933, 193.26277766072317377,

364.27701325326768256, -0.27937980331459933, 157.72987052894848148, 326.94128157610799690, 388.09280021282381767, -0.27937980331459933, 111.18402577739000492, 266.54277820804679777, 346.73233678344666941, 252.57711524406573744, -0.27937980331459933, 57.01639165715037194, 187.85218553479762704, 282.67168803322431359, 224.67611712193377116, -3.14326924385573392, -0.27937980331459933, -0.90010570311670168, 96.17069437543644028, 199.33582061541360986, 182.27846100397448481, -4.57611186272201920, -257.99057047583272606, -1.82557398572480523, 101.69189406672646214, 127.54445573081555665, -4.49732115287934420, -232.30379071310053973, -392.35336998126877006, -2.65962064013652721, 64.23306507635152229, -4.17823235146674676, -190.62136057342786444, -350.97981405658060794, -367.33195935840376478, -3.92525467655692495, -4.21737374407493348, -136.02000312436450713, -288.23032697439572303, -329.76997539886201594, -217.04860831373792962, -4.22506971989652058, -72.29251053978725849, -205.62211432499287866, -270.03888658781465892, -195.10425396920604157, -0.33966732733873783, -3.75470502920668459, -107.71214521643315720, -191.64672875892048864, -159.55913514871488701, -0.33966732733873783, -2.82958247867568247, -100.33839131188909732, -113.28188975877471023, -0.33966732733873783, -2.02984471265764910, -59.20137760431371987, -0.33966732733873783, -1.13006153433920531, -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, -216.94986764694439785, -0.27937980331459933, -194.17815277072924118, -366.46414265432213142, -0.27937980331459933, -58.68817436396830090, -0.27937980331459933, -158.58371851794368013, -328.49641010394003615, -389.37906114078765540, -0.27937980331459933, -112.59894550018677251, -99.47273320793448192, -268.50544005644428580, -349.78532743252520731, -254.30643984757679732, -190.57684701519576720, -106.72241271551203567, -286.77766766087427186, -228.97129601689493938, 0.65888959003758174, -203.43166351223138122, -71.95880175563581815, -188.94812353702479868, -0.54028097490086324, 255.71015351091924117, -135.09478864364308492, -4.18373247132461756, -1.97138030481194448, 228.04006865312763352, 391.09379142082968883, -3.47586424690169338, 65.01418211509854927, 184.65606105019628558, 348.36366144437067760, 366.20261451786950602, 128.93974257358527780, 101.83529783301392513, 283.51551025120562599, 327.60501736958264019, 216.03661549146309540, 200.00439085138680184, 96.43013782801264711, 267.22322299296490655, 193.49255409229277802, -0.33966732733873783, 188.08418865096339800, 57.04436608707574408, 157.96464661316298361, -0.33966732733873783, 111.31922987220724508, -0.33966732733873783, -0.33966732733873783, -0.↩
33966732733873783]

- list testCase.bFieldCorrectValue = [[-0.00000466916987661623140515657, -0.00000469774735703660250055961, -0.00000395939995441359537849948, -0.00000482221010938081362973313, -0.00000396550195293026650080566, -0.00000387749965995507482504874, -0.00000382771914822608855781148, -0.00000292862967710368168300470, -0.00000309944588351366663844040, -0.00000326900335531990481345490, -0.00000454936400457388206747429, -0.00000558752101129238260083512, -0.00000408059138231045771583226, -0.00000355106400521909434142326, -0.00000435087021876521834478800, -0.00000601269263477531150000705, -0.00000422084203782920821688886, -0.00000318190138831836488695379, -0.00000410009973941543825440466, -0.00000596980094276740785646726, -0.00000437558746190408254440234, -0.00000290454751947236613300574, -0.00000382714510184938365401023, -0.00000548811576122437301257333, -0.00000452500225311201280219012, -0.00000289673824026572774027722, -0.00000355797907710025893013046, -0.00000472917096365148288520958, -0.00000464209266810378363380275, -0.00000325927840943272435566430, -0.00000331142973766262746783385, -0.00000391818849209657636467391, -0.00000209963437544594680927643, -0.00000334845606049689288255403, -0.00000442231665001654005299307, -0.00000340778215041781557426582, -0.00000496401499704663627101850, -0.00000371020589188032213470627, -0.00000647174234685046458954343, -0.00000310184058961244829062285, -0.00000741610960273363378910619, -0.00000230211586279070535689746, -0.00000754218056388060709003445, -0.00000166104789262922323326264, -0.00000688055635928999741163627, -0.00000152421125169085425935086, -0.00000570765164247120032554915, -0.00000834218389464975617795476, -0.00000857509678229675104794115, -0.00000171698141817773268018820, -0.00000083362705561108169433266, -0.00000779149266707861698196078, -0.00000061122225394470813803995, -0.00000634459329939409620016319, -0.00000132985936788230726855145, -0.00000474768270446881180025947, -0.00000294559945549423868964008, -0.00000349515548030209205720683, -0.00000506152148848857683034784, -0.00000363192483452629967890547, -0.00000705892632586430391890102, -0.00000280316637579680072320460, -0.00000484372545126640032442291, -0.00000351759447383940304743635, -0.00000278445555434836104581934, -0.00000506790710227895837522742, -0.00000360543134416433626724991, -0.00000723002305488891725015511, -0.00000270154459553407161272231, -0.00000862752206377693032246307, -0.00000151603425631581637699668, -0.00000889681845434635380212227, -0.00000054574943740535989470102, -0.00000807459544423345471292688, -0.00000028768415648587549632195, -0.00000653979801767764028344381, -0.00000104852021517869108351137, -0.00000172984723925869242351476, -0.00000083983432612446153178254, -0.00000843900811589068215474302, -0.00000767582118307908633147649, -0.00000059726856794431001312451, -0.00000625821688852544495012837,

-0.000001284687519605434690504 85, -0.000004693463019647930058576 93, -0.000002865153004397145685826 78,
-0.000003470761878876645504734 97, -0.000004949419569968011847413 41, -0.000003634935900347054723181 36,
-0.000006924696060383603073097 50, -0.000002812678644541278844166 95, -0.000008199362443059459207815 59,
-0.000003171962524546940482986 57, -0.000004713747175559267898895 32, -0.000003715651603631717748402 36,
-0.000003360228058128201781591 80, -0.000006173589524796339052627 79, -0.000003119009788903953190816 48,
-0.000007106838088744919283376 06, -0.000002325287924573052330488 30, -0.000007257567319432155692989 55,
-0.000001671477311721885492592 23, -0.000006646439844438127494139 28, -0.000001494427110945597850732 12,
-0.000005536930259193586047802 15, -0.000002003327756573957827 84, -0.000004316458287959380088992 78,
-0.000005529433366181379469364 40, -0.000005137848765220240516486 1, -0.000002913492218500592313208 02,
-0.000002846558478894211337349 54, -0.000004480448295283521281098 72, -0.000003105741629932859720935 25,
-0.000003766094683489923555699 39, -0.000003674135521447355052886 14, -0.000003200685755163319901570 185,
-0.000004412662744884351176931 10, -0.000003834529390517351181183 57973, -0.000005101301723723616199498 64,
-0.000003572254233953652774409 33, -0.000005518687668104206204293 57, -0.000003209646821722285806244 77,
-0.000003013788779416146341538 86, -0.000004107786858267637268018 42, -0.000003886649395090982258128 92,
-0.000003972274445804966337366 69, -0.000004100691593327836342842 11, -0.000003686432516574610400249 66,
-0.000004243769233644540846952 99, -0.000003520000496158839291098 44, -0.000004371178943848989181643 785,
-0.000003376627843257072685126 44, -0.000004443780700306835564072 92, -0.000003244387453385726021207 97,
-0.000004425754754760034221446 23, -0.000003121464441778340444456 26, -0.000004305061406002007946998 42,
-0.000001625397001290687052922 32, -0.000002010421170774849839644 98, -0.000006025176983159446724813 53,
-0.000005740816591202888841940 97, -0.000002475171262408580079907 59, -0.000004948165877237216171552 57,
-0.000002824910051833316077289 03, -0.000003831068787161905272455 83, -0.000004106338475345775961266 96,
-0.000002707293732265527070896 68, -0.000004620230251661191355090 89, -0.000001885700362389204486434 73,
-0.000005263276692959339155247 33, -0.000001531639762121057001811 27, -0.000005810663830738314987727 37,
-0.000004214427283803097153223 19, -0.000005047567038276975362530 27, -0.000001709896970175021093158 35,
-0.000003682056512467596892742 5, -0.000006104964874039769308507 71, 0.000000147970715399805712407 52,
-0.000006998154097385826055103 14, -0.000000139543256453925005840 17, -0.000007325311268714011280823 22,
-0.000000957449086117795926051 14, -0.000006814847503880925708760 32, -0.000001919098645207326594053 39,
-0.000005462538042366831693268 4, -0.000002659382888561019648448 76, -0.000003582988164699999556671 26,
-0.000008127960838667446259319 00, -0.000007457537766348704991409 37, -0.000000234238167987601168211 21,
-0.000001530497702892756555463 50, -0.000005735090100133847753390 71, -0.000002538748528520541344422 14,
-0.000003362471878848714567928 75, -0.000004277978857843769559853 38, -0.000001009160251116196961387 07,
-0.000005313268141089583862582 92, 0.000000665965933245403488290 42, -0.000006633416101256644790214 35,
0.000001286449913954700983257 92, -0.000007741212825529410768436 90, 0.000000871395950407652559332 63,
-0.000000686976806836211766085 19, 0.000001087106470070684856888 5107, -0.000005372574302659216289191 49,
-0.000006762387901441110924850 07, 0.000001735673372247141482926 88, -0.000007920443808640041069872 24,
0.000001272895962333544302592 24, -0.000008304913740411444737061 18, 0.000000062266992184529590216 48,
-0.000007564410539416802209118 47, -0.000001360573441233412476806 04, -0.000005713940709338972224811 89,
-0.000002477620552432487205092 41, -0.000003185525822041595355317 13, -0.000004285070874239019008832 17,
-0.000000099761910657629347219 19, -0.000001427228406237246719401 23, -0.000007121890788624390749520 65,
-0.000005403874044001729978867 37, -0.000002481624482490750735133 46, -0.000003073674819372230883302 93,
-0.000004232310602584853629661 02, -0.000000777199925254534555820 50, -0.000005212217344728846509167 07,
0.000000853268705209387845752 14, -0.000006467267976458186151970 85, 0.000001450366431284919655866 67,
-0.000007504937291090763591802 55, 0.000001022957800677181251607 08, -0.000007828939574459940902730 58,
-0.000004852333446753194925538 20, -0.000005787573750391326826518 75, 0.000000020287080273935243494 07,
0.000000494123708124895733633 46, -0.000006552513892308689661726 51, 0.000000176526386444774723423 77,
-0.000006767676752540858455193 30, -0.000000683554916541925163217 55, -0.000006192424092432093462169 51,
-0.000001712915954191518598656 24, -0.000004844962781809541334590 57, -0.000002546711824114191281778 44,
-0.000003032326826771812897331 99, -0.000004125421682941469509386 60, -0.000001249268188050931304157 69,
-0.000004900055231256016360420 24, -0.000004100792051189442145528 19, -0.000002167802257612540191332 80,
-0.000002659935093759879472431 36, -0.000003044511846408271801877 12, -0.000003978446178241522450977 18,
-0.000002008865214462643643099 23, -0.000004343371960747792585877 13, -0.000001264973063471340136853 40,
-0.000004820345137294431170111 19, -0.000009704770876797764053115 4, -0.000005199108208945110929364 61,
-0.000001123954920169712695175 01, -0.000005269881142691021274377 695, -0.000001590297779107672852580 53,
-0.000002790076419657463466597 34, -0.000002707920534783716912688 40, -0.000003705861395300735868418 76,
-0.000003639831032048499611622 66, -0.000002670940549635263427431 54, -0.000003547311581062633304968 58,
-0.000002674698768215354493561 50, -0.000003419714389657307361184 00, -0.000002717673792120109748461 43,
-0.000003259265982187504637304 95, -0.000002801549574795857155024 32, -0.000003083134575019942040502 83,
-0.000003811687890361988188879 34, -0.000002918682155814867151366 44, -0.000003758574669448968346373 82,

-0.000003275371435838905081349 87, -0.000002948112853648078168658 60, -0.000002431318723147511385679 30,
-0.000003126679961502833175648 45, -0.000003648708365475053834204 32, -0.000003827816526512429553283 45,
-0.000003183129155410189311398 16, -0.000004313944058749856084794 20, -0.000002607368642943563552788 04,
-0.000004447290998259228003187 95, -0.000002102292897194702951088 65, -0.000004223755108950629635965 02,
-0.000001848759477094594787840 43, -0.000003767219061504040205539 19, -0.000001960451867220521750473 98,
-0.000005923420840851475748943 45, -0.000005544760707924370150411 36, -0.000000814003347248115565701 91,
-0.000000425324422685005126110 83, -0.000004699591704416164490286 81, -0.000000737345699662157190757 71,
-0.000003754035505591927710645 04, -0.000001737684900976831395982 64, -0.000003076109720272052050648 72 847,
-0.000003165602208531349471594 28, -0.000003512825164681814800536 19, -0.000004596921397320425354713 83,
-0.000002701527771238801390441 38, -0.000005604500464185683460084 31, -0.000001687199993214851172440 79,
-0.000003423634737932706773077 05, -0.000003165555164106575179364 50, -0.000003199467932657861860961 40,
-0.000005114769646552445833177 52, -0.000002384952630150790012382 97, -0.000006470135878608972665752 74,
-0.000001082573787397974524103 57, -0.000006914685681239927260075 24, 0.000000031670643635583581195 60,
-0.000006434154519761684247382 98, 0.000000507715209298222229401 54, -0.000005329878432634425243974 09,
0.000000062410153394725731244 98, -0.000004080209874444685043360 86, -0.000001287619385107277441640 81,
0.000000305300963399226811309 37, 0.000000804931633721443997277 14, -0.000006759106531276178780106 44,
-0.000005563573402434483273145 34, 0.000000310536619466591689908 48, -0.000004204891421702167735529 29,
-0.000001158304945279706829906 53, -0.000003203034977046413815985 30, -0.000003231981014917997459149 46,
-0.000003394080338639974761408 63, -0.000005310010027204825310540 89, -0.000002280563636248469533586 57,
-0.000006784693242381760576659 77, -0.000000884776552097538224539 25, -0.000007274133226560342348231 74,
-0.000003268978054311831548653 29, -0.000005161678927616114361952 47, -0.000003428506799238336935137 98,
-0.000002404325456694791921576 65, -0.000006506812171696135195986 17, -0.000001125054147143240300041 59,
-0.000006951900538091945776892 31, -0.000000038109380746886272560 05, -0.000006473838782693360876770 51,
0.000000414566471161729396374 99, -0.000005367828254428663770656 07, -0.000000040386984611623721950 87,
-0.000004110903410176831024986 33, -0.000001380732056537456377710 15, -0.000003183728073491490218310 11,
-0.000005627275284744765789656 80, -0.000004776581892182495043305 39, -0.000000612278458906465059219 61,
-0.000000948270888847624863399 24, -0.000003815478480871763776377 08, -0.000001934438735728849900871 64,
-0.000003112106362082083816687 08 44, -0.000003318697971439304780631 21, -0.000003521997951986014315948 69,
-0.000004704946597670969516812 73, -0.000002738495074569903078875 49, -0.000005688898621621768605146 74,
-0.000001769528991031094350896 57, -0.000006004517856198477860068 13, -0.000000951497156201585765080 38,
-0.000003661100896278850712182 60, -0.000003234185591876674678596 00, -0.000004026008636874376506436 38,
-0.000004469495847771409014577 08, -0.000002724001468378236545539 67, -0.000004586721590835389027304 62,
-0.000002302351486545032412730 11, -0.000004355930598681501865628 54, -0.000002128833052767958072446 46,
-0.000003885539920361719780144 90, -0.000002287528065415136106905 07, -0.000003367620965454054092580 58 81,
-0.000002750529684441294707799 61, -0.000003001205317270282573659 20, -0.000003390656970543113683374 27,
-0.000003914326068653997588988 81, -0.000003867189823242919772985 74, -0.000002837600839175084593070 69,
-0.000002840604460088215984267 35, -0.000003720199167780288467612 42, -0.000002870573199346581707468 64,
-0.000003490914196441135331953 16, -0.000003825832351644726756509 46, -0.000003245016470337282255470 34,
-0.000003818641803553960688288 62, -0.000003048967944455735619121 21, -0.000003847908125245524118163 58,
-0.000002926265125261060909717 68, -0.000003892308380863364904416 87, -0.000002862738197587269642238 56,
-0.000002492694581753588705861 06, -0.000001655482237209619174163 01, -0.000004406155336826438219778 94,
-0.000004975340784626303956589 71, -0.000001287683300810760832579 88, -0.000005486593961721792739652 89,
-0.000001385756011294490192730 30, -0.000005706228818302739558515 45, -0.000001801015181645569025002 26,
-0.000005457598964741178506026 97, -0.000002320718332930050728204 12, -0.000004707444945448636833360 69,
-0.000002743208692611579841867 32, -0.000003616020659077754512405 33, -0.000003992667533797237232255 84 122,
-0.000000943664169711307778902 31, -0.000001893959634366278747664 54, -0.000006824526704582180925199 01,
-0.000005569244589370603003308 27, -0.000002641951482762336413701 443, -0.000003753790329346908617768 12,
-0.000004138060931600643000847 70, -0.000001891613888080865604464 45, -0.000004910989905867787144321 15,
-0.000000511858972162371407152 39, -0.000005940090633514393263399 37, 0.000000066869698384312723880 37,
-0.000006849347666819941143358 76, -0.000000162249097066521752707 46, -0.000007239504929639869315930 42,
0.000000191391818799561805345 25, 0.000000914529967001440018990 81, -0.000006601026404545618444402 21,
-0.000007780046901856716892802 92, 0.000000606924975258649377351 26, -0.000008288517969195862596641 12,
-0.000000407243207521755645005 55, -0.000007768517864930579020074 29, -0.000001632687759934762493114 08,
-0.000006184252258447503232798 11, -0.000002585877388757879148791 99, -0.000003894962601007848762828 95,
-0.000004241965490617003539544 71, -0.000001547524222414936022091 27, -0.000005260361892204286456337 70,
-0.000001584287326956026389354 14, -0.000002587251259447109409157 64, -0.000006477356464652411522243 26,
-0.000004041447503927193620506 91, -0.000004290813773417610484193 36, -0.000001538564664527889685189 31,
-0.000005405471948075501606651 49, 0.000000322772421190406936908 86, -0.000006863814565985649504911 44,

0.00000110565623209020424979579, -0.00000814501678208472028113688, 0.00000078861949552759908613274,
-0.00000870371448507973557073749, -0.00000028510788378058330532662, -0.00000815944672833694236064889,
-0.00000669537711149468838303916, -0.00000789813763548346784327137, 0.00000059111690822940169047717,
0.00000033958897532037827406143, -0.00000843450370123183468468522, -0.00000060731020670712381307903,
-0.00000795474925824088177606606, -0.00000176360881033100408983962, -0.00000642607693187222163444007,
-0.00000264950953861316024627484, -0.00000419727137355453706597883, -0.00000427950624441869564276632,
-0.00000189329076971206080654735, -0.00000532885580003296280786730, -0.00000016218104512392893321314,
-0.00000605647666546202333815807, -0.00000435775685590026293258267, -0.00000421210241176505481092629,
-0.00000276658645189778249273769, -0.00000257663455914345744930628, -0.00000504689462856587593453947,
-0.00000120775748752362999718360, -0.00000612872368050149496745190, -0.00000056728422944763438312402,
-0.00000708742265097004913354441, -0.00000068582099788866186343856, -0.00000753550741748602772706461,
-0.00000133551002728087000193395, -0.00000720199719923842319161842, -0.00000215038968822335375453720,
-0.00000220864124536301706543378, -0.00000214455871493291925708458, -0.00000584699542421314695728662,
-0.00000615773447218937881853483, -0.00000236862926813422162815250, -0.00000603227633872227872430900,
-0.00000269205395604865386263152, -0.00000544191020866243409731056, -0.00000292369433509105852898478,
-0.00000451521056375472740033756, -0.00000410111253397477487728793, -0.00000350115052254179988177425,
-0.00000460709538149641507480967, -0.00000267126967331914903975611, -0.00000525745487178567678444082],[0.↩
00000236689735854612036034493, 0.00005401354211893874220952461, 0.00005361209752247405858861448,
0.00000212634386834270124150520, 0.00021376343909891860873249969, 0.00021601933245713852613754946,
0.00021371351716211045026171111, -0.00021300359426054614703054113, -0.00021061582816840062628259422,
-0.00021061741798744331561309140, -0.00004934873012581498011829007, -0.00004941124532952747715110892,
0.00020639727021737062418772768, 0.00020623915423405027010431145, -0.00009677353186531814817350483,
-0.00009669204184451884575429714, 0.00019274496593224478991485116, 0.00019244750241484230969882252,
-0.00013664227172964252764696125, -0.00013648502089441260243934950, 0.00017129427512268155758649868,
0.00017086687692046981928983840, -0.00016740123771768520926327140, -0.00016724183195677489507241509,
0.00014084176142589512905310689, 0.00014033720852142282046332022, -0.00018914725864201307700164512,
-0.00018903785169325571565573929, 0.00010124684434453062561228703, 0.00010074910169068024355220475,
-0.00020305356881622632094462622, -0.00020301004742418946296539373, 0.00009964288170678496799580193,
0.00005283182990893548782156824, -0.00020276272880443801764627221, -0.00021056322965899858831555214,
0.00000183567463261882781206812, 0.00021360306438548135651162019, -0.00004919220042996906192781653,
0.00020586211499470417700917435, -0.00009608966261168552585539976, 0.00019172569358311930781506305,
-0.00013571607969028911654801872, 0.00016983536798967206228605509, -0.00016652900459506974457056516,
0.00013914990339860717496155151, -0.00018853503281114595440705328, -0.00009509148455064945668849946,
-0.00013447767023880887729200828, 0.00019068224041758288387835019, 0.00016834579280365134334833399,
-0.00016538868811190722638709838, 0.00013744358406513892349037897, -0.00018772671509909759704481935,
0.00009807098094741700583173993, -0.00020235617890487298388239867, 0.00005175681024165314616028444,
-0.00021046484583792980389144178, 0.00000149892188486529660175635, 0.00021344673154304538102016275,
-0.00004876438534439745959691431, 0.00020531907274229157784987965, -0.00020185860419682120025196281,
-0.00021034037193842330644780181, 0.00005053109160962290390042145, 0.00000114154901151524062632807,
0.00021326595474644230443495019, -0.00004822146945023034870321174, 0.00020468795438172604668480847,
-0.00009387570356633688213425504, 0.00018946919872457241326539623, -0.00013297905005305176600639838,
0.00016661659992437002041076877, -0.00016400930448360015218196850, 0.00013546794870449764699137096,
-0.00018674596885885963773975271, 0.00009625906936821321464078233, 0.00018826451889321853911882232,
0.00016490430486719209789495122, -0.00013145884283336355395806205, -0.00016261042982944199416367459,
0.00013351928059236953662096414, -0.00018574983365833599405388121, 0.00009448165801620361033304196,
-0.00020135117527323337675501658, 0.00004934124374795620691447687, -0.00021021156638463859515214560,
0.00000081579080715768989086648, 0.00021308578346141500339111508, -0.00004765240590694399928656938,
0.00020405991050963801490027161, -0.00009263694554953974698557201, 0.00004837466835853511658857920,
0.00000058155701039145530455290, 0.00021293116020920565471634400, -0.00021010043660045379953528466,
-0.00004713469428258448720671023, 0.00020352549070900421979146699, -0.00009155976766124312125894047,
0.00018724422757668106320760870, -0.00013014991383698186564604748, 0.00016346134239983613071003909,
-0.00016140975972011515878271060, 0.00013188813148966749711626933, -0.00018489545185913661227289362,
0.00009300879276891316639174628, -0.00020091557636532487540574621, -0.00012925270539438139002878791,
-0.00016059350334162808765850472, 0.00016249542860370777726253255, 0.00013080809160116255314941502,
-0.00018431712987497476529015483, 0.00009205034882169432243741863, -0.00020062165658016288126604121,
0.00004776862409358900338988280, -0.00021002581909411262750704552, 0.00000047092379963368467166607,
0.00021282322220287654655768161, -0.00004674645129560488617187844, 0.00020316101713698048940141827,
-0.00009080711069766315328768980, 0.00018655538713025239376010200, -0.00021000045739016207665787739,

0.000000468496722975468725434 71, -0.000046570894974512044676984 16, 0.000212776183591962213649886 88,
0.000203017284636241576245910 27, -0.000090508659689697462430620 08, 0.000186294559890174982750080 11,
-0.000128912599494081291182784 32, 0.000162139397234438431549818 92, -0.000160291530754248264098102 42,
0.000130421087905862660054692 74, -0.000184106934719242398140748 31, 0.000091721448271931169766643 67,
-0.000200516910719880100217801 80, 0.000047580422904333251820589 02, -0.000160557719816197100358781 68,
-0.000184300847421255760590075 30, 0.000130766200997182237579161 13, 0.000092041590851325123192853 86,
-0.000200619233900458353173426 60, 0.000047799011715865603639082 08, -0.000210029014512176810929727 78,
0.000000532571303976980690958 70, 0.000212795264389177541351771 96, -0.000046670850250016787761232 07,
0.000203112219705347667739539 50, -0.000090735178138577786606001 90, 0.000186494734899948858821913 67,
-0.000129196355327063495781192 43, 0.000162435842229606639680605 80, 0.000212875970359699450210055 89,
0.000203428421306936594312522 74, -0.000047051586169250888159066 44, -0.000091470201312729388721037 28,
0.000187121525124687927872646 49, -0.000130073492317495011810768 45, 0.000163334936925583942401302 15,
-0.000161358668415655109743336 86, 0.000131784712822059215875605 44, -0.000184872418188891028635253 98,
0.000092952643557140575696247 36, -0.000200913995275991171928636 76, 0.000048376858798901613983606 96,
-0.000210107320454922511963544 16, 0.000000634559435855381146126 54, 0.000133329104367413051306784 37,
0.000094329593865078435093243 78, -0.000185735537609260761881049 83, -0.000201355965797107855233885 61,
0.000049242235424952767071371 53, -0.000210222976609329898801947 77, 0.000000771610574274012064539 59,
0.000213004828755086945447916 32, -0.000047650083777419015805803 39, 0.000203915828729739876962884 23,
-0.000092603441979978340001765 37, 0.000188078223405378422558240 08, -0.000131414566344109873203908 75,
0.000164701630295302340646623 75, -0.000162575441295885285680061 78, -0.000048358240542782781555251 85903,
-0.000093952712733040200554675 28, 0.000204499160763000102040282 12, 0.000189219154298339451098290 30,
-0.000133010853923613869050773 81, 0.000166331494185268286044657 17, -0.000164020307842087647641756 25961,
0.000135174630940144074394837 78, -0.000186756679712913014734104 73, 0.000095981201274371452246587 93,
-0.000201875961792950960570214 30, 0.000050287066748564196905398 05, -0.000210357235307407665341856 35,
0.000000945831546361488028779 70, 0.000213161468552725143861190 09, -0.000187775075555341548413923 47,
-0.000202391314814704703962609 095, 0.000097661762708549853606934 68, 0.000051361120933515946256631 25,
-0.000210487880383046306764119 57, 0.000001142268415573614673906 69, 0.000213321737595187808228221 67484,
-0.000049055358650310537703756 46, 0.000205089184217387615399505 46, -0.000095300402920074772108137 43,
0.000190370555610442862169359 72, -0.000134609317898834126734849 97, 0.000167977443041761651367285 71,
-0.000165465818928934839610392 36, 0.000137043352496085294811925 54, 0.000205596273242615845125089 16,
0.000191356880560719828671817 36, -0.000096429345734682237866602 92, -0.000135953323875821389206050 57,
0.000169387099236767527227369 426, -0.000166680613984285935780249 59, 0.000138646318272511482076400 64,
-0.000188627633186269226452796 98, 0.000099108713805188419386157 93, -0.000202818616221805365248415 98,
0.000052295007860146025980164 35, -0.000210592676573964068462152 53, 0.000001330359001511941881057 45,
0.000213461389402232079538432 32, -0.000049628667999307527359392 77, 0.000100095668511400949962815 43,
0.000052942045806407151960042 15, -0.000203086748763526170769716 52, -0.000210652858558419564397393 15,
0.000001482797231078990299573 80, 0.000213559778799039438152915 23, -0.000049981868773111537396746 82,
0.000205944208740124212462220 31, -0.000097152928638991490413630 29, 0.000192028477701103756367442 84,
-0.000136822024657846930129068 14, 0.000170344496869580697574059 20, -0.000167465770306396846867985 46,
0.000139732856229210159744267 11, -0.000189174562319515025045557 46, -0.000097340819937043174584455 35,
-0.000137064165602774170379246 63, 0.000192285509876484640582119 94, 0.000170706340480553461639093 29,
-0.000167686274376499988483096 85, 0.000140147361084074802270102 44, -0.000189321596988780403453961 34,
0.000100477298597832136271140 73, -0.000203148151925788763043009 83, 0.000053212739632084984727255 405,
-0.000210656118694941364493863 90, 0.000001590473867554816558511 31, 0.000213603000323906870626736 18,
-0.000050042845332859443050008 84, 0.000206081987010186276846902 61, -0.000202986512863565650001382 884,
-0.000210598625469318365653614 84, 0.000053087264801852376497861 78, 0.000001661969305362335990822 40,
0.000213585979126585659079728 96, -0.000049776611175894935004840 20, 0.000205991687493712791371853 45,
-0.000096940750199499544173011 02, 0.000192093693335806091088829 86, -0.000136623010183028434664717 63,
0.000170424908901014638719362 27, -0.000167293129211993189185959 18104, 0.000139830959006838372645636 91,
-0.000189035332686850177357634 76, 0.000100211685893732794583668 72, -0.000135548222352933954524406 57,
-0.000166332456678990107902180 17, 0.000169554305763017427229749 63, 0.000138850911705568015510475 77,
-0.000188349098960462744446753 28, 0.000099361363228582282890189 5583, -0.000202619621994215077425208 20,
0.000052610641167274193718411 68, -0.000210485749353153150360498 39066, 0.000001713373908990238250800 59,
0.000213513193711971978451448 66312, -0.000049198502421013869513686 38, 0.000205691116958947478708881 992,
-0.000095992045901571732322893 00560, 0.000191489246002345131361962 00187, 0.000213397915609302287962001 37,
-0.000210331375162080074728981 89, 0.000001759661147329894187042 84, -0.000048377218822747961219938 46,
0.000205230849964004898175126 69, -0.000094623227002166648770715 98, 0.000190572600814000511076917 73,
-0.000133990044805612300814201 54, 0.000168240309297866292171513 84, -0.000164939163113440466403719 63,

0.00013737511321888575574305669, -0.00018735823757851658355119695, 0.00009807884414871537084174014,
-0.00020209691050995912033755308, 0.00005187996759700072140495189, 0.00016669626682979941695479653,
0.00013564600188451338743517161, -0.00016331554584148575727520214, -0.00018620541242146896178215332,
0.00009658019822995888290558669, -0.00020149203778585097405866289, 0.00005102759603463510423545027,
-0.00021015590352159302920710682, 0.00000181251451853404934958251, 0.00021326009234269244023950163,
-0.00004742244181526253778233976, 0.00020468619001289613560864988, -0.00009302998797134073090571438,
0.00018949218651015582465940712, -0.00013217460481848797510320837, 0.00000188183096733821720918046,
-0.00004646306178227798219142544, 0.00021312320187091661548121901, 0.00020414536535904166366164891,
-0.00009144059856885732679929180, 0.00018842107421555658302442238, -0.00013036638263550800022685938,
0.00016516932778448990284080045, -0.00016169910417474810376131655, 0.00013394203931830443027317656,
-0.00018505853569537650173820742, 0.00009511098788541958836805640, -0.00020089172519817714238771211,
0.00005020161802245859730946947, -0.00020998324462182294566740814, -0.00016032597487864515706929014,
-0.00018408506926736689060693231, 0.00013253533509886359221911778, 0.00009390970411895653872577650,
-0.00020038261517049631003396792, 0.00004954477677215978978237515, -0.00020983726687291497873394386,
0.00000197578821772630547676697, 0.00021301057107211401734671308, -0.00004562627780974468425087207,
0.00020369581070737321592915847, -0.00009008022043937425173479316, 0.00018753014173850509000232578,
-0.00012882727487069587095531997, 0.00016390236616676046358352659, 0.00021294172529952472413747322,
0.00020341065109891282784541910, -0.00004502166004309501721363310, -0.00008913988972122294034920037,
0.00018696181859359240753568066, -0.00012777871683883405525342103, 0.00016309694613150704160624282,
-0.00015939577032451232260247542, 0.00013165059781623895062384333, -0.00018342677777614482836389398,
0.00009317197016279605257822044, -0.00020003820446215492788455403, 0.00004917235129883729257901925,
-0.00020973824458595628230338226, 0.00000209747380690658305900007, 0.00013143059119712788090049427,
0.00009301959721049492247051810, -0.00018317858475078900928889603, -0.00019990783628202538199300498,
0.00004915206393052292354033800, -0.00020969984424572089831695365, 0.00000224057723819562692061454,
0.00021292932978375189549680169, -0.00004472865265418797314431187, 0.00020333744884580098898942124,
-0.00008875081727962814482409820, 0.00018680831799521921822264392, -0.00012737006056659394451040235,
0.00016288256973613317985481730, -0.00015904222592714236392610894, -0.00004478461487772766256567833,
-0.00008896441909145279562567271, 0.00020349091060582237284332097, 0.00018709768610017716130092258,
-0.00012765611819718898720804612, 0.00016329711683072274100089749, -0.00015931336728019438722311896,
0.00013191433264510263648726374, -0.00018337456214367854005070635, 0.00009348155509948895258969664,
-0.00020000939188601177557910726, 0.00004949306625213800757759341, -0.00020972710623493162466514372,
0.00000238983050735444537608729, 0.00021297718916038889478553608, -0.00018398315763743915704844789,
-0.00020032675406499094263902683, 0.00009449216034864956273442438, 0.00005014870894073889949667244,
-0.00020981572132553852100440850, 0.00000252865124875403699013774, 0.00021307960767526475998924484,
-0.00004517421321322968294281405, 0.00020385063315851497875441645, -0.00008974154212259016652562110,
0.00018778976759760101469982907, -0.00012858803279102554682766790, 0.00016428179261177328789912677,
-0.00016016452372189310328527700, 0.00013303259613173861407829646, -0.00009095873254367217101545168,
-0.00013002181463364067626545251, 0.00018878250468945596720997393, 0.00016569080202047639280814317,
-0.00016146599517999967333264921, 0.00013462003300271314703770442, -0.00018491239573189665462704667,
0.00009590359714470447229615890, -0.00020081238310014058756104716, 0.00005102676789639610463571471 8,
-0.00020995270643397084863249202, 0.00000264452492808631948819384, 0.00021322220444996276391205758,
-0.00004583031933930512133099078, 0.00020436418097369337125608635, -0.00020139457054747816238185476,
-0.00021011837168406554765889838, 0.00005199824380140341512730393, 0.00000272280146821860843779368,
0.00021338405786834859704896583, -0.00004665295441673965732816767, 0.00020495498505802614542213591,
-0.00009243352129873707472614702, 0.00018992737085119523830073485, -0.00013174350693034869328942871,
0.00016731337563349006755317183, -0.00016302382379238214318677158, 0.00013643967874164272336204307,
-0.00018602406973527294193650194, 0.00009750611236171419617460576, 0.00019105156639495987505941721,
0.00016890490087334553408887627, -0.00013350205524091196627317624, -0.00016460890798362668775191298,
0.00013821707486982872888948981, -0.00018715448846857901167052562, 0.00009905634124965563037934491,
-0.00020198824848110914710781760, 0.00005291133229075066968152136, -0.00021028927693523695495805836,
0.00000274215455054867476503079, 0.00021354085513318514727779118, -0.00004753000996580814083592587,
0.00020553385861102628408975113, -0.00009395478661697239395125814, 0.00005361925186353092243020033,
0.00000268632338141147337766401, 0.00021366857053165268470577842, -0.00021044173118870139838946368,
-0.00004834422379271559745261139, 0.00020601242450934444377236288, -0.00009530652655494274011482858,
0.00019198380347336301888529209, -0.00013504261160062464952429095, 0.00017022320317893461409999856,
-0.00016598979521232747752490511, 0.00013968135484984135763486535, -0.00018813872729791416139703830,
0.00010031552524288051128284471, -0.00020250774239127889504892555, -0.00013614143289071532218724447,
-0.00016696641590870862350767800, 0.00017106461539639585727259896, 0.00014060608245580221136589172,

-0.0001888352198330815246742064 9, 0.0001010878807277874153695146 3, -0.0002028795988306857968081625 7,
0.0000540089530001394441810491 6, -0.0002105553053265399223847181 8, 0.0000025567534832831971575606 7,
0.0002137471200392704058067944 2, -0.0000489804546699348350219036 0, 0.0002063164712131958309759144 4,
-0.0000962931402029267502137854 8, 0.0001925799746497466663139319 6],[0.0002422142241956319557570543 8,
0.0002351211375509709396777052 7, 0.0002305805786319710642528341 1, 0.0002374875916315655546839913 4,
0.0000364207993591956640010501 3, 0.0000000487615462799760644805 5, 0.0000357377205735378227586737 7,
-0.0000000208828632649373691521 5, 0.0000364306177319406545070910 9, 0.0000356951791195401937732689 7,
0.0002352704418307118440067266 0, 0.0002306333251322870050558505 6, 0.0000735833788797728686635266 1,
0.0000722044630495227859152615 5, 0.0002154290968723453507296244 9, 0.0002111463024388018406857919 8,
0.0001116962960810461238415902 7, 0.0001096003479632281986822645 0, 0.0001857273685521011697893700 7,
0.0001820096959762213548270765 0, 0.0001497705713525325575252028 5, 0.0001469492181445083219155922 6,
0.0001500360341721017646762770 2, 0.0001470179428706962243035560 9, 0.0001854348797551410514116571 0,
0.0001819194066247202003477195 8, 0.0001118901927678630202084991 7, 0.0001096327507685009125960179 4,
0.0002151742998518192879261329 3, 0.0002110602455711169372215563 2, 0.0000736854639396054117948808 8,
0.0000721972234985886413341504 8, 0.0001988274113344844305079073 3, 0.0002171713996453709396260156 6,
0.0000679343190072937021641366 6, 0.0000335873453288910070436111 6, 0.0002236289933713853333822896 3,
0.0000336825636766093156667517 5, 0.0002171310469756153381602065 1, 0.0000680505316879141420325152 7,
0.0001987498216468110720598488 9, 0.0001032902376825338332938519 8, 0.0001712990607415597321943473 4,
0.0001384762374608919128105705 1, 0.0001383514197093156324926133 1, 0.0001714064018702344692392930 4,
0.0001031628079983103348209877 4, 0.0001787158692465807467795319 0, 0.0001540089449117183620970722 4,
0.0000930051610758595272490656 4, 0.0001246718388598578525994706 5, 0.0001243720877810673873924287 3,
0.0001542930682076652185857890 1, 0.0000927311243935130042519501 8, 0.0001789389105262196211134501 1,
0.0000610617467882111755522398 0, 0.0001954025762695806760563255 2, 0.0000301884425360119514416840 8,
0.0002011662203663506739460820 1, 0.0000303339954764597336098110 7, 0.0001952797336970343866375393 6,
0.0000612798140472538342519992 5, 0.0000518446262841833208014248 6, 0.0000256293500447819632824964 6,
0.0001661064240173648241197940 1, 0.0001709609631495680789198266 0, 0.0000258204840014318118719529 5,
0.0001659198248814118146696183 8, 0.0000521515229364738976841099 6, 0.0001518165022405068445652404 1,
0.0000791378802867689321335828 4, 0.0001308066663486571817257703 2, 0.0001060620171846635791026297 6,
0.0001056199223639653497026566 5, 0.0001312309599735850284586702 3, 0.0000787405674602860715426480 6,
0.0001521529356570801240402385 2, 0.0000622204068311170638630458 5, 0.0000833595954992759477150676 7,
0.0001025863279702575838522177 2, 0.0000828178103014302056158330 0, 0.0001031019683656121991525284 1,
0.0000617303227987699122073644 9, 0.0001194929869466503010453153 9, 0.0000406378966869730428625173 7,
0.0001304032617439559820198685 7, 0.0000200854566673863485092195 5, 0.0001341707869393013994631369 5,
0.0000203154402231195115115122869 3, 0.0001301792888758991010699633 8, 0.0000410161954941623096808356 3,
0.0001190871489139969308579045 1, 0.0000896612951166579933912070 8, 0.0000922050575450178390088384 9,
0.0000140305624026271624212947 4, 0.0000137698927469755589967541 7, 0.0000894286661487138660655421 6,
0.0000283023400772633142315724 8, 0.0000817848658957147508444945 9, 0.0000429040594542518121420719 7,
0.0000704331018335910063543972 6, 0.0000574384700085536642759527 5, 0.0000568429989292959915643598 5,
0.0000709876532152992642619310 2, 0.0000423547626265096056854040 , 0.0000822125942406185498826168 3,
0.0000278725384342885966547191 7, 0.0000355824237684743306794552 0, 0.0000286934276260946750469575 7,
0.0000293008405307374635222029 5, 0.0000361307262171594887408161 7, 0.0000213582367278607547207789 5,
0.0000417536032540618889594141 7, 0.0000140389797633587990648498 8, 0.0000454533400417786025583116 7,
0.0000692532427039974761851914 , 0.0000466801596698611208227994 2, 0.0000072076712886117065155733 8,
0.0000452350084328073052388971 6, 0.0000144999005127954385053123 6, 0.0000413431465856834383766974 0,
0.0000219344847247146686707031 1, -0.0000001853627547618876580709 7, -0.0000006355508831214799005159 3,
-0.0000006910511212498215423310 8, 0.0000001093706514781205700636 3, 0.0000001411258539840001720040 3,
-0.0000006781948455703794525188 3, 0.0000001222355231533376071311 2, -0.0000006252075513818507458596 2,
0.0000000373396731326234099717 7, -0.0000005498547900381242401033 0, -0.0000001140612108938774679956 7,
-0.0000004533550523421004894015 7, -0.0000003087297503186224864260 4, -0.0000003317327149449809965472 6,
-0.0000004996161987516389804984 7, -0.0000297639121433423091672015 9, -0.0000222433248848979186400136 3,
-0.0000363382479527750079846028 2, -0.0000423353826789528642693047 3, -0.0000146882262038876563379997 6,
-0.0000464041629645930850866716 9, -0.0000072891362689723049604921 2, -0.0000478969234639896001529298 1,
-0.0000069911197922881611148664 8, -0.0000465646067181385082175276 7, -0.0000142202735765586503133024 0,
-0.0000426538618534731911544760 6, -0.0000216897174122289768246826 8, -0.0000367955250531364896622414 7,
-0.0000292181713376094043569612 0, -0.0000138205387214580815860923 1, -0.0000280308042319632119698802 3,
-0.0000906099076149304092763631 1, -0.0000829642422149054682050176 0, -0.0000426596177741803283416967 3,
-0.0000715375834656913327930238 8, -0.0000573353344030572256760994 2, -0.0000578278079582850951614360 3,
-0.0000711405871458394619549769 9, -0.0000431762107190321361006694 9, -0.0000826985160618557107840842 5,

-0.00002847984121712703668954751, -0.00009048023917856811966595881, -0.00001411327712725095190880210,
-0.00009327121660228933602409390, -0.00010318331096309616562445299, -0.00011984926043979200348168351,
-0.00006244976420704460807684089, -0.00004117762498992887174997096, -0.00013103893622691932877022403,
-0.00002039581194293668963161623, -0.00013502106123139932542659425, -0.00002011614821639516964232354,
-0.00013114060303592856016052792, -0.00004075872687014014603135159, -0.00012006568355820456989353773,
-0.00006197987877876467933870724, -0.00010352074553398723177032037, -0.00008323192631180004077799672,
-0.00008366647472881513195215597, -0.00016661982395304531716188901, -0.00015254720284414819103484262,
-0.00005191459148860947298053173, -0.00007890803977945857377099270, -0.00013152454305630307466838669,
-0.00010591450193005250970598730, -0.00010629131245839624965877218, -0.00013124084510660762969806725,
-0.00007932537509100781797973662, -0.00015237219495052033743469344, -0.00005229449907647272581491180,
-0.00016654075739959696661300625, -0.00002589553811933865517776535, -0.00017156478127045330598493544,
-0.00002563586837672157138761503, -0.00009315588308157499923371769, -0.00006140378828223462081940809,
-0.00017902977521352931135692477, -0.00019563886207072810237417304, -0.00003040125774207811000515722,
-0.00020151795372538009896642985, -0.00003016758948722044825696206, -0.00019570217940511938615154308,
-0.00006106996966507975167781641, -0.00017917336493616733169957633, -0.00009279481635922343226927089,
-0.00015447997982178347862840251, -0.00012451474238563082835504203, -0.00012483612445501407950051997,
-0.00015424205974566204940118253, -0.00006787379414957882630982794, -0.00010310846865362845376553474,
-0.00019892317488940359734329866, -0.00017150610400596397794176218, -0.00013832132833937183760859846,
-0.00013858878543495013356198764, -0.00017130791711928515627834380, -0.00010341001070076125513755305,
-0.00019880316650244890031885847, -0.00006815557561437242427900690, -0.00021722025818499458046755002,
-0.00003373987825876595495486546, -0.00022373374742836631742098907, -0.00003353732141415467825438373,
-0.00021727331389793976402884101, -0.00021093144903028035572374599, -0.00023045009102277296854366373,
-0.00007229013674218415913801800, -0.00003578305871748360640150280, -0.00023734955778653999964970811,
-0.00003561583216870238619965877, -0.00023049492413110690388136637, -0.00007206567010067558643263141,
-0.00021103047189652274348223293, -0.00010945482579816623021438243, -0.00018194432559134879529884332,
-0.00014680650550965147586868209, -0.00014701869566590132997439822, -0.00018178460549161468487455728,
-0.00010969288485332497659185264, -0.00021502053205313794098998970, -0.00018538753811050880798694251,
-0.00011159209960883873999359928, -0.00014964640440136669271437020, -0.00014979879388047950741075232,
-0.00018526987130899826320892809, -0.00011176162833692815856342917, -0.00021494487333164816319705515,
-0.00007364803063842092860618455, -0.00023481149497267908045350371, -0.00003645214620836874505008934,
-0.00024183043309658122915067890, -0.00003632358939450872130317791, -0.00023484697328218957779188969,
-0.00007348570070165847160307410, -0.00007217637459090645165541572, -0.00003572123132083137171362622,
-0.00023013193893893477723171237, -0.00023699750321345817473395190, -0.00003563375048566358671781398,
-0.00023015541463671364361825988, -0.00007208051415000035354492536, -0.00021073441288608306246922519,
-0.00010944015028969240280881658, -0.00018169947874284282326486084, -0.00014673388428673416199361779,
-0.00014681974367343732994044270, -0.00018163039591441909516994491, -0.00010953530158290421248328678,
-0.00021068714758319437405260921, -0.00017102143724016463524964515, -0.00013819485027111137070456037,
-0.00013818324391302191597209748, -0.00017100862949847816488375962, -0.00010309824844230417221178919,
-0.00019832459509211473967088546, -0.00006793096195648446161251960, -0.00021659394815229924581567733,
-0.00003361817217347092407982515, -0.00022303888842768569570132464, -0.00003357316702557160831288385,
-0.00021660238469981582358492300, -0.00006790521451532017442685035, -0.00019833757494762031077170650,
-0.00010308355091668143076882869, -0.00003022129276137249811150232, -0.00003022354191583747934791299,
-0.00020050224236140999287125730, -0.00019471799236716816209127878, -0.00006112119459283027955113360,
-0.00017831210229449142310245713, -0.00009276843179126121057172238, -0.00015376580633080783044874806,
-0.00012432618713406346899058574, -0.00012425530965746002867722275, -0.00015381792982871233870997840,
-0.00009269686204412431063809352, -0.00017834029739462084170401046, -0.00006107419556489151579291991,
-0.00019472833737965098690678167, -0.00010569872516909659565954166, -0.00013072542721564734541361830,
-0.00010553637997225742131343629, -0.00007872981769881417493574943, -0.00015151256669335006464251203,
-0.00005186886194571795803904060, -0.00016538868205494963790729013, -0.00002566757221347684439992885,
-0.00017026715794004254264648601, -0.00002570711508851162356808877, -0.00016535368945022483060276752,
-0.00005198983398827113598661409, -0.00015143334855863968611659753, -0.00007889283784995311619037822,
-0.00013059750346209480532932989, -0.00013350340066229854428352108, -0.00012964271048749101411523554,
-0.00002020415769903493376528567, -0.00004086236607941256832737809, -0.00011873569373521749984509016,
-0.00006199115823709242520996726, -0.00010240612573972078156413212, -0.00008301960174735155861998337,
-0.00008275595700465254278121313, -0.00010262405433081328446145997, -0.00006173266849323452069939894,
-0.00011888040906379310635964525, -0.00004066808207705533453587068, -0.00012971161161790919678048206,
-0.00002012517505528861871131122, -0.00005678709604919412133865880, -0.00004235736489269485043670377,
-0.00007059611100592111604737355, -0.00008170085455448483993407677, -0.00002790183370216623858113547,

-0.00008907037541158730050913639, -0.00001380924169836618856728634, -0.00009162394555755865672899002,
-0.00001392380876832649396824312, -0.00008895525178005084677296255, -0.00002816626159222587871540193,
-0.00008147217911830019747721626, -0.00004271309183722462904809342, -0.00007027162350779519084300467,
-0.00005716150568599052202330146, -0.00000710720186852305037012324, -0.00001438873875637678012161420,
-0.00004484880680013963766446239, -0.00004106903905102492186420318, -0.00002179834921734077519159083,
-0.00003542469340099802210504801, -0.00002911664329704440472052114, -0.00002862548975263022965165143,
-0.00003587046457447661982828602, -0.00002134765509186273144603616, -0.00004139993132217327894339584,
-0.00001406054587053311242389464, -0.00004502285445157880360358971, -0.00000696249189633010152450073,
-0.00004623291093189804202342325, 0.00000022467911527007665704246, 0.00000048167492606297729647319,
0.00000048899518627313372006463, 0.00000032363462959153753230231, 0.00000074706668727336450951479,
0.00000015186105810516095130798, 0.00000093369933049566224796936, -0.00000001597525931478165870129,
0.00000098850066969786042197772, -0.00000005809251968019942385924, 0.00000092577002949726541889861,
-0.00000004820078760242361379804, 0.00000079797623325687385482019, 0.00000004145243264645911705269,
0.00000064737491214713786505830, 0.00004680035744017245062957230, 0.00004289967437104083856956940,
0.00001427646083623986728212713, 0.00002170491470029656898136673, 0.00003700333685930434426828231,
0.00002919834367635558365671258, 0.00002990495330871639463946043, 0.00003631130908792697039319877,
0.00002231216647983441349811437, 0.00004234585953618262148560533, 0.00001469727005698481946843227,
0.00004649210278299842238718120, 0.00000726022599652357248197409, 0.00004807255989783239198730527,
0.00000707774193715610308293238, 0.00004328101073928273493252733, 0.00002850699106410683686201567,
0.00008259509056876351233006384, 0.00009046504309814210227636000, 0.00001408917733785442160082246,
0.00009337934338277425647904112, 0.00001390172785790030196692232, 0.00009082805648641118900082914,
0.00002806563960136081002519695, 0.00008323702796946372977921619, 0.00004262872853756081065234110,
0.00007179579171989889567560145, 0.00005724008582118376498634044, 0.00005801929525829632245314746,
0.00007101209043807231804674301, 0.00012038333834286814294761009, 0.00010383306764068984972583520,
0.00006192240669891687482838394, 0.00008309469525665875077810596, 0.00008390578792607722500955514,
0.00010300121812758039553832579, 0.00006258715875818359006960145, 0.00011969106390390556713971409,
0.00004122096533532975401806220, 0.00013098303125910060315181815, 0.00002037599858503987373427313,
0.00013511635305782260002734063, 0.00002019407080782172699169597, 0.00013137857324053720670109513,
0.00004078093529010286559435133, 0.00005234954471178215960457808, 0.00002587883134899454044136613,
0.00016649349418400413627011636, 0.00017168185862034981243803844, 0.00002571317762172624241329491,
0.00016689317445149211391504374, 0.00005193466288569443666412601, 0.00015290857304592109277077694,
0.00007884675684891602855716014, 0.00013188042058524072496113189, 0.00010577246912910297216552785,
0.00010656630312335742445401660, 0.00013105468082139063346232277, 0.00007948646572211354763108637,
0.00015221385474413457556883134, 0.00009275223431934548432714616, 0.00012440355572374795184269203,
0.00015485901300309637775888538, 0.00012512830010899064921668644, 0.00015409710537769833295709820,
0.00009332802808153728359348567, 0.00017891747244875335883310119, 0.00006146429238704890509435719,
0.00019563624679433259164612613, 0.00003038600629161280648467096, 0.00020167483423115019087329403,
0.00003024702323752094363881228, 0.00019600983424611289154294858, 0.00006109853669347578390691478,
0.00017956355425588266033418150, 0.00021729386764661110095428476, 0.00022394623173360359274369769,
0.00003362128005204847520015141, 0.00003372421628176038894833880, 0.00021761287056678890782304703,
0.00006792007336981402225700027, 0.00019932597152122770358534387, 0.00010310376552645847273229074,
0.00017188644896625095018563578, 0.00013827097185156560221752642, 0.00013887814927907385104816074,
0.00017124231788059795437810517, 0.00010357934222891635389022374, 0.00019877640983226803924992232,
0.00006821455257841429445538062, 0.00018230925383158519165367750, 0.00014728808845052273408977872,
0.00014683821948869725635029859, 0.00018182656923916606147434427, 0.00010984671095598225567298994,
0.00021102242965774522750511955, 0.00007234097598022849436374959, 0.00023063008265034889185222944,
0.00003576519103914519034058406, 0.00023763783581554187422126001, 0.00003570603085499142020356120,
0.00023087139189595877854579986, 0.00007213616985492927720455963, 0.00021143688368655628919551281,
0.00010950145260021515328328207]]

8.21 testCase.py

Go to the documentation of this file.

```
00001
00002 from readMesh import CylindricMeshGiven
00003 from defineTargetField import TargetFieldGiven
00004 from sensitivityMatrix import getSensitivityMatrix
```

```
00005 from resistanceMatrix import getResistanceMatrix
00006 from streamFunctionOptimization import streamFunctionOptimization
00007 from Tester import Tester
00008 import numpy as np
00009
00010 meshFile = "cylinder_radius500mm_length1500mm.stl"
00011 targetMeshFile = "sphere_radius150mm.stl"
00012 gaussOrder = 2
00013 tikonovFac = 100
00014 specificConductivityMaterial = 1.8000*10**-8
00015 conducterThickness = 0.005
00016 materialFactor = specificConductivityMaterial/conducterThickness
00017
00018 Test = Tester()
00019 Mesh = CylindricMeshGiven(meshFile)
00020 TargetSphere = TargetFieldGiven(targetMeshFile,1)
00021 sensitivityMatrix = getSensitivityMatrix(Test,Mesh,TargetSphere,gaussOrder)
00022 resistanceMatrix = getResistanceMatrix(Test,Mesh,materialFactor)
00023 BField,SFOpt =
       streamFunctionOptimization(Test,Mesh,TargetSphere,sensitivityMatrix,resistanceMatrix,tikonovFac)
00024
00025 # SF Optimization
00026
00027 def test_finalSF():
00028     assert np.array_equal(np.array(SFOpt),np.array(SFCorrectValue))
00029
00030 def test_bFieldGeneratedByOptSF():
00031     assert np.array_equal(BField,np.array(bFieldCorrectValue))
00032
00033 def test_reducedSF():
00034     assert np.array_equal(Test.reducedSF,np.array(reducedSFCorrect))
00035
00036 # Sensitivity Matrix
00037
00038 def test_gaußLegendre():
00039     assert np.array_equal(Test.gaußLegendre,np.array(gaußLegendreCorrect))
00040
00041 def test_WeightsGauss():
00042     assert np.array_equal(Test.calcWeightsGauss,np.array(calcWeightsGaussCorrect))
00043
00044 # Resistance Matrix
00045
00046 def test_matElementsShouldGetValue():
00047     assert np.array_equal(Test.matElementsShouldGetValue,np.array(matElementsShouldGetValueCorrect))
00048
00049
00050 matElementsShouldGetValueCorrect = [0.0000000000, 1.0000000000, 2.0000000000, 3.0000000000,
       4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000, 9.0000000000, 10.0000000000,
       11.0000000000, 12.0000000000, 13.0000000000, 14.0000000000, 15.0000000000, 16.0000000000,
       17.0000000000, 18.0000000000, 19.0000000000, 20.0000000000, 21.0000000000, 22.0000000000,
       23.0000000000, 24.0000000000, 25.0000000000, 26.0000000000, 27.0000000000, 28.0000000000,
       29.0000000000, 30.0000000000, 31.0000000000, 32.0000000000, 33.0000000000, 34.0000000000,
       35.0000000000, 36.0000000000, 37.0000000000, 38.0000000000, 39.0000000000, 40.0000000000,
       41.0000000000, 42.0000000000, 43.0000000000, 44.0000000000, 45.0000000000, 46.0000000000,
       47.0000000000, 48.0000000000, 49.0000000000, 50.0000000000, 51.0000000000, 52.0000000000,
       53.0000000000, 54.0000000000, 55.0000000000, 56.0000000000, 57.0000000000, 58.0000000000,
       59.0000000000, 60.0000000000, 61.0000000000, 62.0000000000, 63.0000000000, 64.0000000000,
       65.0000000000, 66.0000000000, 67.0000000000, 68.0000000000, 69.0000000000, 70.0000000000,
       71.0000000000, 72.0000000000, 73.0000000000, 74.0000000000, 75.0000000000, 76.0000000000,
       77.0000000000, 78.0000000000, 79.0000000000, 80.0000000000, 81.0000000000, 82.0000000000,
       83.0000000000, 84.0000000000, 85.0000000000, 86.0000000000, 87.0000000000, 88.0000000000,
       89.0000000000, 90.0000000000, 91.0000000000, 92.0000000000, 93.0000000000, 94.0000000000,
       95.0000000000, 96.0000000000, 97.0000000000, 98.0000000000, 99.0000000000, 100.0000000000,
       101.0000000000, 102.0000000000, 103.0000000000, 104.0000000000, 105.0000000000, 106.0000000000,
       107.0000000000, 108.0000000000, 109.0000000000, 110.0000000000, 111.0000000000, 112.0000000000,
       113.0000000000, 114.0000000000, 115.0000000000, 116.0000000000, 117.0000000000, 118.0000000000,
       119.0000000000, 120.0000000000, 121.0000000000, 122.0000000000, 123.0000000000, 124.0000000000,
       125.0000000000, 126.0000000000, 127.0000000000, 128.0000000000, 129.0000000000, 130.0000000000,
       131.0000000000, 132.0000000000, 133.0000000000, 134.0000000000, 135.0000000000, 136.0000000000,
       137.0000000000, 138.0000000000, 139.0000000000, 140.0000000000, 141.0000000000, 142.0000000000,
       143.0000000000, 144.0000000000, 145.0000000000, 146.0000000000, 147.0000000000, 148.0000000000,
       149.0000000000, 150.0000000000, 151.0000000000, 152.0000000000, 153.0000000000, 154.0000000000,
       155.0000000000, 156.0000000000, 157.0000000000, 158.0000000000, 159.0000000000, 160.0000000000,
       161.0000000000, 162.0000000000, 163.0000000000, 164.0000000000, 165.0000000000, 166.0000000000,
       167.0000000000, 168.0000000000, 169.0000000000, 170.0000000000, 171.0000000000, 172.0000000000,
       173.0000000000, 174.0000000000, 175.0000000000, 176.0000000000, 177.0000000000, 178.0000000000,
       179.0000000000, 180.0000000000, 181.0000000000, 182.0000000000, 183.0000000000, 184.0000000000,
       185.0000000000, 186.0000000000, 187.0000000000, 188.0000000000, 189.0000000000, 190.0000000000,
       191.0000000000, 192.0000000000, 193.0000000000, 194.0000000000, 195.0000000000, 196.0000000000,
       197.0000000000, 198.0000000000, 199.0000000000, 200.0000000000, 201.0000000000, 202.0000000000,
       203.0000000000, 204.0000000000, 205.0000000000, 206.0000000000, 207.0000000000, 208.0000000000,
       209.0000000000, 210.0000000000, 211.0000000000, 212.0000000000, 213.0000000000, 214.0000000000,
       215.0000000000, 216.0000000000, 217.0000000000, 218.0000000000, 219.0000000000, 220.0000000000,
       221.0000000000, 222.0000000000, 223.0000000000, 224.0000000000, 225.0000000000, 226.0000000000,
       227.0000000000, 228.0000000000, 229.0000000000, 230.0000000000, 231.0000000000, 232.0000000000,
       233.0000000000, 234.0000000000, 235.0000000000, 236.0000000000, 237.0000000000, 238.0000000000,
       239.0000000000, 240.0000000000, 241.0000000000, 242.0000000000, 243.0000000000, 244.0000000000,
```

```
245.0000000000, 246.0000000000, 247.0000000000, 248.0000000000, 249.0000000000, 250.0000000000,
251.0000000000, 252.0000000000, 253.0000000000, 254.0000000000, 255.0000000000, 256.0000000000,
257.0000000000, 258.0000000000, 259.0000000000, 260.0000000000, 261.0000000000, 262.0000000000,
263.0000000000, 1.0000000000, 2.0000000000, 0.0000000000, 6.0000000000, 8.0000000000, 9.0000000000,
3.0000000000, 2.0000000000, 7.0000000000, 3.0000000000, 1.0000000000, 5.0000000000, 4.0000000000,
1.0000000000, 9.0000000000, 4.0000000000, 2.0000000000, 12.0000000000, 10.0000000000, 18.0000000000,
12.0000000000, 2.0000000000, 5.0000000000, 3.0000000000, 10.0000000000, 4.0000000000, 11.0000000000,
14.0000000000, 8.0000000000, 15.0000000000, 7.0000000000, 1.0000000000, 14.0000000000, 6.0000000000,
0.0000000000, 1.0000000000, 6.0000000000, 15.0000000000, 16.0000000000, 17.0000000000, 1.0000000000,
23.0000000000, 9.0000000000, 12.0000000000, 17.0000000000, 1.0000000000, 3.0000000000, 8.0000000000,
18.0000000000, 78.0000000000, 4.0000000000, 5.0000000000, 81.0000000000, 77.0000000000,
11.0000000000, 77.0000000000, 10.0000000000, 18.0000000000, 9.0000000000, 17.0000000000,
4.0000000000, 3.0000000000, 19.0000000000, 21.0000000000, 7.0000000000, 14.0000000000, 15.0000000000,
22.0000000000, 21.0000000000, 13.0000000000, 7.0000000000, 6.0000000000, 14.0000000000,
29.0000000000, 8.0000000000, 6.0000000000, 22.0000000000, 23.0000000000, 8.0000000000, 30.0000000000,
17.0000000000, 23.0000000000, 25.0000000000, 24.0000000000, 9.0000000000, 12.0000000000,
25.0000000000, 16.0000000000, 27.0000000000, 8.0000000000, 19.0000000000, 26.0000000000,
12.0000000000, 10.0000000000, 4.0000000000, 81.0000000000, 85.0000000000, 19.0000000000,
26.0000000000, 33.0000000000, 27.0000000000, 12.0000000000, 18.0000000000, 17.0000000000,
21.0000000000, 13.0000000000, 14.0000000000, 20.0000000000, 28.0000000000, 13.0000000000,
144.0000000000, 143.0000000000, 22.0000000000, 15.0000000000, 28.0000000000, 14.0000000000,
35.0000000000, 29.0000000000, 21.0000000000, 29.0000000000, 30.0000000000, 8.0000000000,
15.0000000000, 16.0000000000, 37.0000000000, 30.0000000000, 31.0000000000, 25.0000000000,
16.0000000000, 32.0000000000, 16.0000000000, 17.0000000000, 31.0000000000, 27.0000000000,
24.0000000000, 85.0000000000, 18.0000000000, 19.0000000000, 33.0000000000, 90.0000000000,
33.0000000000, 32.0000000000, 17.0000000000, 25.0000000000, 19.0000000000, 34.0000000000,
151.0000000000, 22.0000000000, 35.0000000000, 147.0000000000, 144.0000000000, 21.0000000000,
30.0000000000, 35.0000000000, 36.0000000000, 22.0000000000, 23.0000000000, 15.0000000000,
29.0000000000, 43.0000000000, 37.0000000000, 23.0000000000, 24.0000000000, 36.0000000000,
16.0000000000, 37.0000000000, 24.0000000000, 32.0000000000, 39.0000000000, 38.0000000000,
25.0000000000, 39.0000000000, 41.0000000000, 31.0000000000, 25.0000000000, 34.0000000000,
27.0000000000, 19.0000000000, 27.0000000000, 34.0000000000, 90.0000000000, 40.0000000000,
26.0000000000, 41.0000000000, 40.0000000000, 47.0000000000, 32.0000000000, 27.0000000000,
33.0000000000, 36.0000000000, 29.0000000000, 151.0000000000, 22.0000000000, 42.0000000000,
28.0000000000, 42.0000000000, 29.0000000000, 35.0000000000, 30.0000000000, 43.0000000000,
43.0000000000, 44.0000000000, 24.0000000000, 30.0000000000, 31.0000000000, 38.0000000000,
31.0000000000, 39.0000000000, 37.0000000000, 46.0000000000, 45.0000000000, 44.0000000000,
46.0000000000, 48.0000000000, 41.0000000000, 38.0000000000, 31.0000000000, 32.0000000000,
34.0000000000, 90.0000000000, 33.0000000000, 96.0000000000, 47.0000000000, 103.0000000000,
34.0000000000, 54.0000000000, 47.0000000000, 48.0000000000, 32.0000000000, 39.0000000000,
35.0000000000, 43.0000000000, 156.0000000000, 36.0000000000, 162.0000000000, 49.0000000000,
151.0000000000, 50.0000000000, 36.0000000000, 37.0000000000, 56.0000000000, 44.0000000000,
30.0000000000, 42.0000000000, 49.0000000000, 51.0000000000, 38.0000000000, 43.0000000000,
57.0000000000, 45.0000000000, 37.0000000000, 50.0000000000, 52.0000000000, 51.0000000000,
46.0000000000, 44.0000000000, 38.0000000000, 39.0000000000, 38.0000000000, 52.0000000000,
53.0000000000, 45.0000000000, 48.0000000000, 34.0000000000, 103.0000000000, 40.0000000000,
54.0000000000, 41.0000000000, 39.0000000000, 53.0000000000, 55.0000000000, 41.0000000000,
54.0000000000, 46.0000000000, 61.0000000000, 42.0000000000, 162.0000000000, 43.0000000000,
56.0000000000, 57.0000000000, 63.0000000000, 56.0000000000, 44.0000000000, 43.0000000000,
45.0000000000, 44.0000000000, 52.0000000000, 58.0000000000, 57.0000000000, 65.0000000000,
45.0000000000, 46.0000000000, 51.0000000000, 60.0000000000, 59.0000000000, 58.0000000000,
53.0000000000, 46.0000000000, 55.0000000000, 52.0000000000, 60.0000000000, 48.0000000000,
48.0000000000, 109.0000000000, 61.0000000000, 47.0000000000, 103.0000000000, 115.0000000000,
41.0000000000, 60.0000000000, 62.0000000000, 61.0000000000, 48.0000000000, 53.0000000000,
50.0000000000, 49.0000000000, 43.0000000000, 63.0000000000, 169.0000000000, 162.0000000000,
58.0000000000, 51.0000000000, 63.0000000000, 64.0000000000, 50.0000000000, 44.0000000000,
51.0000000000, 57.0000000000, 65.0000000000, 52.0000000000, 64.0000000000, 67.0000000000,
72.0000000000, 52.0000000000, 66.0000000000, 65.0000000000, 60.0000000000, 52.0000000000,
53.0000000000, 59.0000000000, 67.0000000000, 62.0000000000, 55.0000000000, 121.0000000000,
68.0000000000, 62.0000000000, 115.0000000000, 48.0000000000, 55.0000000000, 54.0000000000,
55.0000000000, 68.0000000000, 61.0000000000, 60.0000000000, 69.0000000000, 67.0000000000,
64.0000000000, 56.0000000000, 57.0000000000, 50.0000000000, 175.0000000000, 70.0000000000,
169.0000000000, 58.0000000000, 57.0000000000, 71.0000000000, 70.0000000000, 63.0000000000,
65.0000000000, 52.0000000000, 64.0000000000, 58.0000000000, 59.0000000000, 72.0000000000,
75.0000000000, 71.0000000000, 59.0000000000, 72.0000000000, 66.0000000000, 62.0000000000,
59.0000000000, 60.0000000000, 127.0000000000, 69.0000000000, 73.0000000000, 121.0000000000,
61.0000000000, 62.0000000000, 68.0000000000, 67.0000000000, 62.0000000000, 63.0000000000,
71.0000000000, 64.0000000000, 181.0000000000, 187.0000000000, 175.0000000000, 74.0000000000,
74.0000000000, 64.0000000000, 70.0000000000, 65.0000000000, 75.0000000000, 75.0000000000,
65.0000000000, 59.0000000000, 68.0000000000, 69.0000000000, 76.0000000000, 75.0000000000,
71.0000000000, 187.0000000000, 193.0000000000, 70.0000000000, 74.0000000000, 65.0000000000,
76.0000000000, 71.0000000000, 193.0000000000, 74.0000000000, 10.0000000000, 78.0000000000,
79.0000000000, 10.0000000000, 84.0000000000, 77.0000000000, 81.0000000000, 80.0000000000,
79.0000000000, 83.0000000000, 80.0000000000, 82.0000000000, 78.0000000000, 79.0000000000,
82.0000000000, 84.0000000000, 78.0000000000, 18.0000000000, 10.0000000000, 84.0000000000,
85.0000000000, 78.0000000000, 88.0000000000, 86.0000000000, 80.0000000000, 83.0000000000,
87.0000000000, 84.0000000000, 79.0000000000, 86.0000000000, 82.0000000000, 82.0000000000,
81.0000000000, 89.0000000000, 88.0000000000, 80.0000000000, 85.0000000000, 78.0000000000,
81.0000000000, 18.0000000000, 84.0000000000, 26.0000000000, 90.0000000000, 89.0000000000,
91.0000000000, 87.0000000000, 92.0000000000, 82.0000000000, 93.0000000000, 82.0000000000,
91.0000000000, 88.0000000000, 86.0000000000, 89.0000000000, 93.0000000000, 87.0000000000,
82.0000000000, 94.0000000000, 84.0000000000, 90.0000000000, 84.0000000000, 88.0000000000,
95.0000000000, 85.0000000000, 94.0000000000, 89.0000000000, 40.0000000000, 102.0000000000,
26.0000000000, 33.0000000000, 95.0000000000, 85.0000000000, 96.0000000000, 93.0000000000,
92.0000000000, 97.0000000000, 99.0000000000, 86.0000000000, 87.0000000000, 98.0000000000,
91.0000000000, 97.0000000000, 87.0000000000, 99.0000000000, 100.0000000000, 91.0000000000,
94.0000000000, 88.0000000000, 105.0000000000, 89.0000000000, 88.0000000000, 106.0000000000,
```

100.0000000000, 101.0000000000, 93.0000000000, 95.0000000000, 101.0000000000, 102.0000000000,
90.0000000000, 89.0000000000, 94.0000000000, 103.0000000000, 108.0000000000, 40.0000000000,
90.0000000000, 102.0000000000, 209.0000000000, 210.0000000000, 104.0000000000, 92.0000000000,
213.0000000000, 99.0000000000, 98.0000000000, 91.0000000000, 97.0000000000, 209.0000000000,
104.0000000000, 105.0000000000, 93.0000000000, 97.0000000000, 91.0000000000, 94.0000000000,
93.0000000000, 105.0000000000, 106.0000000000, 102.0000000000, 106.0000000000, 94.0000000000,
95.0000000000, 112.0000000000, 107.0000000000, 101.0000000000, 107.0000000000, 96.0000000000,
113.0000000000, 95.0000000000, 90.0000000000, 108.0000000000, 108.0000000000, 109.0000000000,
114.0000000000, 47.0000000000, 96.0000000000, 40.0000000000, 54.0000000000, 110.0000000000,
99.0000000000, 213.0000000000, 97.0000000000, 219.0000000000, 105.0000000000, 100.0000000000,
106.0000000000, 99.0000000000, 104.0000000000, 93.0000000000, 110.0000000000, 116.0000000000,
111.0000000000, 105.0000000000, 100.0000000000, 112.0000000000, 111.0000000000, 117.0000000000,
94.0000000000, 101.0000000000, 113.0000000000, 118.0000000000, 102.0000000000, 101.0000000000,
112.0000000000, 103.0000000000, 113.0000000000, 114.0000000000, 102.0000000000, 96.0000000000,
114.0000000000, 115.0000000000, 103.0000000000, 54.0000000000, 120.0000000000, 116.0000000000,
219.0000000000, 105.0000000000, 225.0000000000, 104.0000000000, 105.0000000000, 116.0000000000,
117.0000000000, 106.0000000000, 106.0000000000, 107.0000000000, 101.0000000000, 118.0000000000,
117.0000000000, 123.0000000000, 118.0000000000, 108.0000000000, 107.0000000000, 114.0000000000,
102.0000000000, 119.0000000000, 113.0000000000, 120.0000000000, 108.0000000000, 109.0000000000,
119.0000000000, 103.0000000000, 120.0000000000, 121.0000000000, 61.0000000000, 54.0000000000,
109.0000000000, 117.0000000000, 122.0000000000, 110.0000000000, 225.0000000000, 230.0000000000,
111.0000000000, 105.0000000000, 116.0000000000, 123.0000000000, 122.0000000000, 111.0000000000,
112.0000000000, 106.0000000000, 123.0000000000, 107.0000000000, 112.0000000000, 119.0000000000,
124.0000000000, 113.0000000000, 114.0000000000, 130.0000000000, 125.0000000000, 124.0000000000,
113.0000000000, 120.0000000000, 118.0000000000, 131.0000000000, 115.0000000000, 114.0000000000,
126.0000000000, 125.0000000000, 121.0000000000, 109.0000000000, 119.0000000000, 61.0000000000,
132.0000000000, 115.0000000000, 126.0000000000, 120.0000000000, 127.0000000000, 68.0000000000,
123.0000000000, 116.0000000000, 230.0000000000, 235.0000000000, 117.0000000000, 128.0000000000,
122.0000000000, 117.0000000000, 112.0000000000, 124.0000000000, 118.0000000000, 133.0000000000,
128.0000000000, 129.0000000000, 129.0000000000, 118.0000000000, 123.0000000000, 130.0000000000,
119.0000000000, 130.0000000000, 120.0000000000, 119.0000000000, 131.0000000000, 131.0000000000,
121.0000000000, 120.0000000000, 132.0000000000, 73.0000000000, 68.0000000000, 121.0000000000,
133.0000000000, 235.0000000000, 240.0000000000, 122.0000000000, 123.0000000000, 130.0000000000,
123.0000000000, 133.0000000000, 137.0000000000, 124.0000000000, 134.0000000000, 124.0000000000,
134.0000000000, 135.0000000000, 125.0000000000, 119.0000000000, 129.0000000000, 131.0000000000,
136.0000000000, 132.0000000000, 139.0000000000, 135.0000000000, 130.0000000000, 125.0000000000,
120.0000000000, 126.0000000000, 126.0000000000, 121.0000000000, 131.0000000000, 127.0000000000,
240.0000000000, 128.0000000000, 137.0000000000, 123.0000000000, 129.0000000000, 137.0000000000,
129.0000000000, 130.0000000000, 138.0000000000, 135.0000000000, 130.0000000000, 141.0000000000,
138.0000000000, 134.0000000000, 139.0000000000, 131.0000000000, 131.0000000000, 132.0000000000,
138.0000000000, 140.0000000000, 240.0000000000, 129.0000000000, 133.0000000000, 134.0000000000,
250.0000000000, 245.0000000000, 137.0000000000, 141.0000000000, 135.0000000000, 134.0000000000,
140.0000000000, 136.0000000000, 131.0000000000, 135.0000000000, 255.0000000000, 138.0000000000,
141.0000000000, 142.0000000000, 137.0000000000, 250.0000000000, 139.0000000000, 140.0000000000,
135.0000000000, 138.0000000000, 140.0000000000, 141.0000000000, 21.0000000000, 144.0000000000,
20.0000000000, 146.0000000000, 143.0000000000, 28.0000000000, 21.0000000000, 146.0000000000,
147.0000000000, 146.0000000000, 149.0000000000, 143.0000000000, 149.0000000000, 145.0000000000,
144.0000000000, 150.0000000000, 147.0000000000, 143.0000000000, 150.0000000000, 28.0000000000,
151.0000000000, 146.0000000000, 144.0000000000, 149.0000000000, 145.0000000000, 145.0000000000,
152.0000000000, 148.0000000000, 153.0000000000, 154.0000000000, 146.0000000000, 150.0000000000,
154.0000000000, 155.0000000000, 147.0000000000, 149.0000000000, 151.0000000000, 160.0000000000,
146.0000000000, 156.0000000000, 42.0000000000, 35.0000000000, 155.0000000000, 28.0000000000,
147.0000000000, 150.0000000000, 148.0000000000, 149.0000000000, 153.0000000000, 157.0000000000,
158.0000000000, 159.0000000000, 154.0000000000, 152.0000000000, 149.0000000000, 149.0000000000,
159.0000000000, 153.0000000000, 150.0000000000, 166.0000000000, 160.0000000000, 160.0000000000,
161.0000000000, 156.0000000000, 151.0000000000, 150.0000000000, 155.0000000000, 162.0000000000,
161.0000000000, 151.0000000000, 168.0000000000, 42.0000000000, 164.0000000000, 153.0000000000,
158.0000000000, 152.0000000000, 157.0000000000, 159.0000000000, 165.0000000000, 153.0000000000,
164.0000000000, 154.0000000000, 158.0000000000, 165.0000000000, 153.0000000000, 171.0000000000,
166.0000000000, 167.0000000000, 154.0000000000, 155.0000000000, 150.0000000000, 166.0000000000,
161.0000000000, 155.0000000000, 168.0000000000, 167.0000000000, 156.0000000000, 173.0000000000,
160.0000000000, 49.0000000000, 42.0000000000, 168.0000000000, 56.0000000000, 156.0000000000,
169.0000000000, 164.0000000000, 157.0000000000, 165.0000000000, 163.0000000000, 214.0000000000,
215.0000000000, 170.0000000000, 157.0000000000, 158.0000000000, 164.0000000000, 171.0000000000,
170.0000000000, 159.0000000000, 176.0000000000, 158.0000000000, 172.0000000000, 159.0000000000,
177.0000000000, 171.0000000000, 154.0000000000, 167.0000000000, 160.0000000000, 173.0000000000,
172.0000000000, 178.0000000000, 166.0000000000, 160.0000000000, 161.0000000000, 162.0000000000,
173.0000000000, 156.0000000000, 174.0000000000, 161.0000000000, 179.0000000000, 169.0000000000,
63.0000000000, 56.0000000000, 174.0000000000, 175.0000000000, 168.0000000000, 162.0000000000,
176.0000000000, 215.0000000000, 222.0000000000, 164.0000000000, 227.0000000000, 165.0000000000,
159.0000000000, 165.0000000000, 182.0000000000, 176.0000000000, 166.0000000000, 177.0000000000,
183.0000000000, 178.0000000000, 177.0000000000, 166.0000000000, 167.0000000000, 178.0000000000,
179.0000000000, 167.0000000000, 168.0000000000, 161.0000000000, 184.0000000000, 179.0000000000,
180.0000000000, 168.0000000000, 175.0000000000, 169.0000000000, 185.0000000000, 180.0000000000,
186.0000000000, 174.0000000000, 169.0000000000, 181.0000000000, 70.0000000000, 63.0000000000,
227.0000000000, 165.0000000000, 182.0000000000, 170.0000000000, 171.0000000000, 171.0000000000,
183.0000000000, 182.0000000000, 166.0000000000, 172.0000000000, 188.0000000000, 183.0000000000,
184.0000000000, 172.0000000000, 167.0000000000, 173.0000000000, 168.0000000000, 190.0000000000,
174.0000000000, 184.0000000000, 173.0000000000, 185.0000000000, 175.0000000000, 186.0000000000,
174.0000000000, 185.0000000000, 175.0000000000, 192.0000000000, 186.0000000000, 70.0000000000,
187.0000000000, 171.0000000000, 176.0000000000, 227.0000000000, 237.0000000000, 232.0000000000,
177.0000000000, 188.0000000000, 189.0000000000, 184.0000000000, 188.0000000000, 172.0000000000,
178.0000000000, 177.0000000000, 189.0000000000, 178.0000000000, 179.0000000000, 190.0000000000,
183.0000000000, 173.0000000000, 174.0000000000, 190.0000000000, 179.0000000000, 186.0000000000,
180.0000000000, 191.0000000000, 191.0000000000, 175.0000000000, 192.0000000000, 185.0000000000,
180.0000000000, 181.0000000000, 193.0000000000, 70.0000000000, 74.0000000000, 192.0000000000,
181.0000000000, 189.0000000000, 194.0000000000, 182.0000000000, 237.0000000000, 177.0000000000,

```
183.0000000000, 242.0000000000, 188.0000000000, 190.0000000000, 195.0000000000, 184.0000000000,
183.0000000000, 194.0000000000, 189.0000000000, 185.0000000000, 200.0000000000, 196.0000000000,
179.0000000000, 195.0000000000, 184.0000000000, 191.0000000000, 196.0000000000, 192.0000000000,
201.0000000000, 197.0000000000, 185.0000000000, 186.0000000000, 190.0000000000, 186.0000000000,
181.0000000000, 193.0000000000, 198.0000000000, 197.0000000000, 191.0000000000, 187.0000000000,
198.0000000000, 187.0000000000, 74.0000000000, 192.0000000000, 199.0000000000, 189.0000000000,
188.0000000000, 242.0000000000, 195.0000000000, 199.0000000000, 200.0000000000, 203.0000000000,
190.0000000000, 194.0000000000, 189.0000000000, 191.0000000000, 190.0000000000, 201.0000000000,
200.0000000000, 198.0000000000, 192.0000000000, 205.0000000000, 201.0000000000, 191.0000000000,
202.0000000000, 202.0000000000, 192.0000000000, 197.0000000000, 242.0000000000, 247.0000000000,
203.0000000000, 252.0000000000, 195.0000000000, 194.0000000000, 195.0000000000, 190.0000000000,
203.0000000000, 204.0000000000, 201.0000000000, 196.0000000000, 207.0000000000, 196.0000000000,
197.0000000000, 191.0000000000, 200.0000000000, 204.0000000000, 205.0000000000, 197.0000000000,
205.0000000000, 206.0000000000, 204.0000000000, 195.0000000000, 200.0000000000, 252.0000000000,
257.0000000000, 199.0000000000, 203.0000000000, 201.0000000000, 200.0000000000, 207.0000000000,
206.0000000000, 207.0000000000, 197.0000000000, 201.0000000000, 204.0000000000, 257.0000000000,
203.0000000000, 207.0000000000, 208.0000000000, 261.0000000000, 206.0000000000, 201.0000000000,
208.0000000000, 204.0000000000, 206.0000000000, 261.0000000000, 211.0000000000, 97.0000000000,
210.0000000000, 211.0000000000, 97.0000000000, 212.0000000000, 213.0000000000, 218.0000000000,
209.0000000000, 216.0000000000, 210.0000000000, 217.0000000000, 212.0000000000, 223.0000000000,
210.0000000000, 217.0000000000, 211.0000000000, 218.0000000000, 224.0000000000, 104.0000000000,
218.0000000000, 210.0000000000, 219.0000000000, 97.0000000000, 163.0000000000, 164.0000000000,
221.0000000000, 215.0000000000, 164.0000000000, 222.0000000000, 214.0000000000, 221.0000000000,
170.0000000000, 217.0000000000, 220.0000000000, 223.0000000000, 221.0000000000, 212.0000000000,
220.0000000000, 216.0000000000, 211.0000000000, 223.0000000000, 224.0000000000, 213.0000000000,
212.0000000000, 210.0000000000, 228.0000000000, 224.0000000000, 225.0000000000, 104.0000000000,
110.0000000000, 229.0000000000, 213.0000000000, 214.0000000000, 221.0000000000, 217.0000000000,
214.0000000000, 223.0000000000, 215.0000000000, 226.0000000000, 217.0000000000, 220.0000000000,
222.0000000000, 170.0000000000, 215.0000000000, 226.0000000000, 227.0000000000, 221.0000000000,
221.0000000000, 228.0000000000, 217.0000000000, 218.0000000000, 212.0000000000, 226.0000000000,
233.0000000000, 228.0000000000, 213.0000000000, 229.0000000000, 219.0000000000, 218.0000000000,
219.0000000000, 229.0000000000, 110.0000000000, 116.0000000000, 230.0000000000, 234.0000000000,
223.0000000000, 228.0000000000, 222.0000000000, 221.0000000000, 231.0000000000, 227.0000000000,
231.0000000000, 232.0000000000, 176.0000000000, 226.0000000000, 222.0000000000, 170.0000000000,
182.0000000000, 224.0000000000, 233.0000000000, 231.0000000000, 226.0000000000, 218.0000000000,
223.0000000000, 236.0000000000, 233.0000000000, 234.0000000000, 219.0000000000, 225.0000000000,
224.0000000000, 235.0000000000, 122.0000000000, 225.0000000000, 234.0000000000, 116.0000000000,
226.0000000000, 236.0000000000, 228.0000000000, 232.0000000000, 227.0000000000, 231.0000000000,
227.0000000000, 182.0000000000, 236.0000000000, 237.0000000000, 234.0000000000, 236.0000000000,
224.0000000000, 238.0000000000, 229.0000000000, 241.0000000000, 228.0000000000, 225.0000000000,
238.0000000000, 230.0000000000, 243.0000000000, 229.0000000000, 235.0000000000, 233.0000000000,
239.0000000000, 239.0000000000, 240.0000000000, 128.0000000000, 122.0000000000, 234.0000000000,
230.0000000000, 231.0000000000, 228.0000000000, 232.0000000000, 237.0000000000, 241.0000000000,
233.0000000000, 182.0000000000, 242.0000000000, 236.0000000000, 232.0000000000, 188.0000000000,
241.0000000000, 243.0000000000, 246.0000000000, 233.0000000000, 234.0000000000, 241.0000000000,
244.0000000000, 248.0000000000, 243.0000000000, 234.0000000000, 235.0000000000, 240.0000000000,
137.0000000000, 249.0000000000, 128.0000000000, 235.0000000000, 133.0000000000, 239.0000000000,
245.0000000000, 244.0000000000, 242.0000000000, 233.0000000000, 236.0000000000, 237.0000000000,
238.0000000000, 246.0000000000, 241.0000000000, 194.0000000000, 199.0000000000, 247.0000000000,
237.0000000000, 251.0000000000, 188.0000000000, 246.0000000000, 234.0000000000, 248.0000000000,
239.0000000000, 251.0000000000, 238.0000000000, 246.0000000000, 248.0000000000, 249.0000000000,
253.0000000000, 240.0000000000, 239.0000000000, 249.0000000000, 137.0000000000, 254.0000000000,
240.0000000000, 250.0000000000, 241.0000000000, 238.0000000000, 251.0000000000, 242.0000000000,
243.0000000000, 251.0000000000, 199.0000000000, 252.0000000000, 242.0000000000, 251.0000000000,
243.0000000000, 239.0000000000, 244.0000000000, 253.0000000000, 244.0000000000, 254.0000000000,
258.0000000000, 253.0000000000, 240.0000000000, 245.0000000000, 137.0000000000, 140.0000000000,
254.0000000000, 245.0000000000, 255.0000000000, 259.0000000000, 253.0000000000, 243.0000000000,
252.0000000000, 256.0000000000, 242.0000000000, 247.0000000000, 248.0000000000, 246.0000000000,
251.0000000000, 203.0000000000, 199.0000000000, 247.0000000000, 256.0000000000, 257.0000000000,
260.0000000000, 260.0000000000, 256.0000000000, 258.0000000000, 244.0000000000, 248.0000000000,
249.0000000000, 251.0000000000, 258.0000000000, 245.0000000000, 259.0000000000, 249.0000000000,
250.0000000000, 140.0000000000, 142.0000000000, 250.0000000000, 260.0000000000, 253.0000000000,
251.0000000000, 252.0000000000, 252.0000000000, 203.0000000000, 263.0000000000, 261.0000000000,
206.0000000000, 260.0000000000, 263.0000000000, 262.0000000000, 249.0000000000, 260.0000000000,
254.0000000000, 253.0000000000, 259.0000000000, 250.0000000000, 255.0000000000, 258.0000000000,
254.0000000000, 256.0000000000, 252.0000000000, 253.0000000000, 258.0000000000, 263.0000000000,
257.0000000000, 263.0000000000, 206.0000000000, 257.0000000000, 258.0000000000, 259.0000000000,
262.0000000000, 260.0000000000, 258.0000000000, 257.0000000000],[0.0000000000, 1.0000000000,
2.0000000000, 3.0000000000, 4.0000000000, 5.0000000000, 6.0000000000, 7.0000000000, 8.0000000000,
9.0000000000, 10.0000000000, 11.0000000000, 12.0000000000, 13.0000000000, 14.0000000000,
15.0000000000, 16.0000000000, 17.0000000000, 18.0000000000, 19.0000000000, 20.0000000000,
21.0000000000, 22.0000000000, 23.0000000000, 24.0000000000, 25.0000000000, 26.0000000000,
27.0000000000, 28.0000000000, 29.0000000000, 30.0000000000, 31.0000000000, 32.0000000000,
33.0000000000, 34.0000000000, 35.0000000000, 36.0000000000, 37.0000000000, 38.0000000000,
39.0000000000, 40.0000000000, 41.0000000000, 42.0000000000, 43.0000000000, 44.0000000000,
45.0000000000, 46.0000000000, 47.0000000000, 48.0000000000, 49.0000000000, 50.0000000000,
51.0000000000, 52.0000000000, 53.0000000000, 54.0000000000, 55.0000000000, 56.0000000000,
57.0000000000, 58.0000000000, 59.0000000000, 60.0000000000, 61.0000000000, 62.0000000000,
63.0000000000, 64.0000000000, 65.0000000000, 66.0000000000, 67.0000000000, 68.0000000000,
69.0000000000, 70.0000000000, 71.0000000000, 72.0000000000, 73.0000000000, 74.0000000000,
75.0000000000, 76.0000000000, 77.0000000000, 78.0000000000, 79.0000000000, 80.0000000000,
81.0000000000, 82.0000000000, 83.0000000000, 84.0000000000, 85.0000000000, 86.0000000000,
87.0000000000, 88.0000000000, 89.0000000000, 90.0000000000, 91.0000000000, 92.0000000000,
93.0000000000, 94.0000000000, 95.0000000000, 96.0000000000, 97.0000000000, 98.0000000000,
99.0000000000, 100.0000000000, 101.0000000000, 102.0000000000, 103.0000000000, 104.0000000000,
105.0000000000, 106.0000000000, 107.0000000000, 108.0000000000, 109.0000000000, 110.0000000000,
111.0000000000, 112.0000000000, 113.0000000000, 114.0000000000, 115.0000000000, 116.0000000000,
```

```
117.0000000000, 118.0000000000, 119.0000000000, 120.0000000000, 121.0000000000, 122.0000000000,
123.0000000000, 124.0000000000, 125.0000000000, 126.0000000000, 127.0000000000, 128.0000000000,
129.0000000000, 130.0000000000, 131.0000000000, 132.0000000000, 133.0000000000, 134.0000000000,
135.0000000000, 136.0000000000, 137.0000000000, 138.0000000000, 139.0000000000, 140.0000000000,
141.0000000000, 142.0000000000, 143.0000000000, 144.0000000000, 145.0000000000, 146.0000000000,
147.0000000000, 148.0000000000, 149.0000000000, 150.0000000000, 151.0000000000, 152.0000000000,
153.0000000000, 154.0000000000, 155.0000000000, 156.0000000000, 157.0000000000, 158.0000000000,
159.0000000000, 160.0000000000, 161.0000000000, 162.0000000000, 163.0000000000, 164.0000000000,
165.0000000000, 166.0000000000, 167.0000000000, 168.0000000000, 169.0000000000, 170.0000000000,
171.0000000000, 172.0000000000, 173.0000000000, 174.0000000000, 175.0000000000, 176.0000000000,
177.0000000000, 178.0000000000, 179.0000000000, 180.0000000000, 181.0000000000, 182.0000000000,
183.0000000000, 184.0000000000, 185.0000000000, 186.0000000000, 187.0000000000, 188.0000000000,
189.0000000000, 190.0000000000, 191.0000000000, 192.0000000000, 193.0000000000, 194.0000000000,
195.0000000000, 196.0000000000, 197.0000000000, 198.0000000000, 199.0000000000, 200.0000000000,
201.0000000000, 202.0000000000, 203.0000000000, 204.0000000000, 205.0000000000, 206.0000000000,
207.0000000000, 208.0000000000, 209.0000000000, 210.0000000000, 211.0000000000, 212.0000000000,
213.0000000000, 214.0000000000, 215.0000000000, 216.0000000000, 217.0000000000, 218.0000000000,
219.0000000000, 220.0000000000, 221.0000000000, 222.0000000000, 223.0000000000, 224.0000000000,
225.0000000000, 226.0000000000, 227.0000000000, 228.0000000000, 229.0000000000, 230.0000000000,
231.0000000000, 232.0000000000, 233.0000000000, 234.0000000000, 235.0000000000, 236.0000000000,
237.0000000000, 238.0000000000, 239.0000000000, 240.0000000000, 241.0000000000, 242.0000000000,
243.0000000000, 244.0000000000, 245.0000000000, 246.0000000000, 247.0000000000, 248.0000000000,
249.0000000000, 250.0000000000, 251.0000000000, 252.0000000000, 253.0000000000, 254.0000000000,
255.0000000000, 256.0000000000, 257.0000000000, 258.0000000000, 259.0000000000, 260.0000000000,
261.0000000000, 262.0000000000, 263.0000000000, 0.0000000000, 0.0000000000, 1.0000000000,
1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 1.0000000000, 2.0000000000,
2.0000000000, 2.0000000000, 2.0000000000, 3.0000000000, 3.0000000000, 3.0000000000, 3.0000000000,
3.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000, 4.0000000000,
5.0000000000, 5.0000000000, 5.0000000000, 6.0000000000, 6.0000000000, 6.0000000000, 6.0000000000,
6.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 7.0000000000, 8.0000000000, 8.0000000000,
8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 8.0000000000, 9.0000000000, 9.0000000000,
9.0000000000, 9.0000000000, 9.0000000000, 10.0000000000, 10.0000000000, 10.0000000000, 10.0000000000,
10.0000000000, 10.0000000000, 10.0000000000, 11.0000000000, 11.0000000000, 12.0000000000,
12.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 12.0000000000, 13.0000000000,
13.0000000000, 13.0000000000, 14.0000000000, 14.0000000000, 14.0000000000, 14.0000000000,
14.0000000000, 14.0000000000, 15.0000000000, 15.0000000000, 15.0000000000, 15.0000000000,
15.0000000000, 15.0000000000, 16.0000000000, 16.0000000000, 16.0000000000, 16.0000000000,
16.0000000000, 16.0000000000, 17.0000000000, 17.0000000000, 17.0000000000, 17.0000000000,
17.0000000000, 17.0000000000, 17.0000000000, 18.0000000000, 18.0000000000, 18.0000000000,
18.0000000000, 18.0000000000, 18.0000000000, 18.0000000000, 19.0000000000, 19.0000000000,
19.0000000000, 19.0000000000, 19.0000000000, 19.0000000000, 20.0000000000, 20.0000000000,
21.0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 21.0000000000, 21.0000000000,
21.0000000000, 22.0000000000, 22.0000000000, 22.0000000000, 22.0000000000, 22.0000000000,
22.0000000000, 23.0000000000, 23.0000000000, 23.0000000000, 23.0000000000, 23.0000000000,
24.0000000000, 24.0000000000, 24.0000000000, 24.0000000000, 24.0000000000, 25.0000000000,
25.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 25.0000000000, 26.0000000000,
26.0000000000, 26.0000000000, 26.0000000000, 26.0000000000, 27.0000000000, 27.0000000000,
27.0000000000, 27.0000000000, 27.0000000000, 27.0000000000, 28.0000000000, 28.0000000000,
28.0000000000, 28.0000000000, 28.0000000000, 28.0000000000, 29.0000000000, 29.0000000000,
29.0000000000, 29.0000000000, 29.0000000000, 29.0000000000, 30.0000000000, 30.0000000000,
30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 30.0000000000, 31.0000000000,
31.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 31.0000000000, 32.0000000000,
32.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 32.0000000000, 33.0000000000,
33.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 33.0000000000, 34.0000000000,
34.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 34.0000000000, 35.0000000000,
35.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 35.0000000000, 36.0000000000,
36.0000000000, 36.0000000000, 36.0000000000, 36.0000000000, 37.0000000000, 37.0000000000,
37.0000000000, 37.0000000000, 37.0000000000, 37.0000000000, 38.0000000000, 38.0000000000,
38.0000000000, 38.0000000000, 38.0000000000, 38.0000000000, 39.0000000000, 39.0000000000,
39.0000000000, 39.0000000000, 39.0000000000, 39.0000000000, 40.0000000000, 40.0000000000,
40.0000000000, 40.0000000000, 40.0000000000, 40.0000000000, 41.0000000000, 41.0000000000,
41.0000000000, 41.0000000000, 41.0000000000, 41.0000000000, 42.0000000000, 42.0000000000,
42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 42.0000000000, 43.0000000000,
43.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.0000000000, 43.0000000000,
43.0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 44.0000000000, 44.0000000000,
44.0000000000, 44.0000000000, 45.0000000000, 45.0000000000, 45.0000000000, 45.0000000000,
45.0000000000, 46.0000000000, 46.0000000000, 46.0000000000, 46.0000000000, 46.0000000000,
46.0000000000, 47.0000000000, 47.0000000000, 47.0000000000, 47.0000000000, 47.0000000000,
48.0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 48.0000000000, 48.0000000000,
48.0000000000, 49.0000000000, 49.0000000000, 49.0000000000, 49.0000000000, 50.0000000000,
50.0000000000, 50.0000000000, 50.0000000000, 50.0000000000, 51.0000000000, 51.0000000000,
51.0000000000, 51.0000000000, 51.0000000000, 52.0000000000, 52.0000000000, 52.0000000000,
52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 52.0000000000, 53.0000000000,
53.0000000000, 53.0000000000, 53.0000000000, 53.0000000000, 54.0000000000, 54.0000000000,
54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 54.0000000000, 55.0000000000,
55.0000000000, 55.0000000000, 55.0000000000, 55.0000000000, 56.0000000000, 56.0000000000,
56.0000000000, 56.0000000000, 56.0000000000, 56.0000000000, 57.0000000000, 57.0000000000,
57.0000000000, 57.0000000000, 57.0000000000, 57.0000000000, 58.0000000000, 58.0000000000,
58.0000000000, 58.0000000000, 58.0000000000, 59.0000000000, 59.0000000000, 59.0000000000,
59.0000000000, 59.0000000000, 59.0000000000, 60.0000000000, 60.0000000000, 60.0000000000,
60.0000000000, 60.0000000000, 60.0000000000, 61.0000000000, 61.0000000000, 61.0000000000,
61.0000000000, 61.0000000000, 61.0000000000, 61.0000000000, 62.0000000000, 62.0000000000,
62.0000000000, 62.0000000000, 62.0000000000, 62.0000000000, 63.0000000000, 63.0000000000,
63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 63.0000000000, 64.0000000000,
64.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 64.0000000000, 65.0000000000,
65.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.0000000000, 65.0000000000,
66.0000000000, 66.0000000000, 67.0000000000, 67.0000000000, 67.0000000000, 67.0000000000,
```

```
68.0000000000, 68.0000000000, 68.0000000000, 68.0000000000, 68.0000000000, 68.0000000000,
69.0000000000, 69.0000000000, 69.0000000000, 70.0000000000, 70.0000000000, 70.0000000000,
70.0000000000, 70.0000000000, 70.0000000000, 70.0000000000, 71.0000000000, 71.0000000000,
71.0000000000, 71.0000000000, 71.0000000000, 72.0000000000, 72.0000000000, 72.0000000000,
73.0000000000, 73.0000000000, 74.0000000000, 74.0000000000, 74.0000000000, 74.0000000000,
74.0000000000, 74.0000000000, 75.0000000000, 75.0000000000, 75.0000000000, 75.0000000000,
76.0000000000, 76.0000000000, 77.0000000000, 77.0000000000, 77.0000000000, 78.0000000000,
78.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 78.0000000000, 79.0000000000,
79.0000000000, 79.0000000000, 79.0000000000, 80.0000000000, 80.0000000000, 80.0000000000,
80.0000000000, 81.0000000000, 81.0000000000, 81.0000000000, 81.0000000000, 81.0000000000,
82.0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 82.0000000000, 82.0000000000,
82.0000000000, 83.0000000000, 83.0000000000, 84.0000000000, 84.0000000000, 84.0000000000,
84.0000000000, 84.0000000000, 84.0000000000, 84.0000000000, 85.0000000000, 85.0000000000,
85.0000000000, 85.0000000000, 85.0000000000, 85.0000000000, 86.0000000000, 86.0000000000,
86.0000000000, 86.0000000000, 87.0000000000, 87.0000000000, 87.0000000000, 87.0000000000,
87.0000000000, 88.0000000000, 88.0000000000, 88.0000000000, 88.0000000000, 88.0000000000,
88.0000000000, 89.0000000000, 89.0000000000, 89.0000000000, 89.0000000000, 89.0000000000,
89.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000, 90.0000000000,
90.0000000000, 90.0000000000, 90.0000000000, 91.0000000000, 91.0000000000, 91.0000000000,
91.0000000000, 91.0000000000, 91.0000000000, 92.0000000000, 92.0000000000, 92.0000000000,
93.0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 93.0000000000, 93.0000000000,
93.0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 94.0000000000, 94.0000000000,
94.0000000000, 94.0000000000, 95.0000000000, 95.0000000000, 95.0000000000, 95.0000000000,
95.0000000000, 96.0000000000, 96.0000000000, 96.0000000000, 96.0000000000, 96.0000000000,
97.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.0000000000, 97.0000000000,
97.0000000000, 97.0000000000, 98.0000000000, 98.0000000000, 99.0000000000, 99.0000000000,
99.0000000000, 99.0000000000, 99.0000000000, 100.0000000000, 100.0000000000, 100.0000000000,
100.0000000000, 101.0000000000, 101.0000000000, 101.0000000000, 101.0000000000, 101.0000000000,
101.0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000, 102.0000000000,
102.0000000000, 102.0000000000, 103.0000000000, 103.0000000000, 103.0000000000, 103.0000000000,
103.0000000000, 103.0000000000, 103.0000000000, 104.0000000000, 104.0000000000, 104.0000000000,
104.0000000000, 104.0000000000, 104.0000000000, 105.0000000000, 105.0000000000, 105.0000000000,
105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 105.0000000000, 106.0000000000,
106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000, 106.0000000000,
107.0000000000, 107.0000000000, 107.0000000000, 107.0000000000, 107.0000000000, 108.0000000000,
108.0000000000, 108.0000000000, 108.0000000000, 108.0000000000, 109.0000000000, 109.0000000000,
109.0000000000, 109.0000000000, 109.0000000000, 110.0000000000, 110.0000000000, 110.0000000000,
110.0000000000, 110.0000000000, 111.0000000000, 111.0000000000, 111.0000000000, 111.0000000000,
112.0000000000, 112.0000000000, 112.0000000000, 112.0000000000, 112.0000000000, 112.0000000000,
113.0000000000, 113.0000000000, 113.0000000000, 113.0000000000, 113.0000000000, 113.0000000000,
114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000, 114.0000000000,
115.0000000000, 115.0000000000, 115.0000000000, 115.0000000000, 115.0000000000, 116.0000000000,
116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000, 116.0000000000,
117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000, 117.0000000000,
118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000, 118.0000000000,
119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000, 119.0000000000,
119.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000, 120.0000000000,
120.0000000000, 120.0000000000, 120.0000000000, 121.0000000000, 121.0000000000, 121.0000000000,
121.0000000000, 121.0000000000, 121.0000000000, 121.0000000000, 122.0000000000, 122.0000000000,
122.0000000000, 122.0000000000, 122.0000000000, 122.0000000000, 123.0000000000, 123.0000000000,
123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000, 123.0000000000,
124.0000000000, 124.0000000000, 124.0000000000, 124.0000000000, 124.0000000000, 125.0000000000,
125.0000000000, 125.0000000000, 125.0000000000, 126.0000000000, 126.0000000000, 126.0000000000,
126.0000000000, 127.0000000000, 127.0000000000, 127.0000000000, 128.0000000000, 128.0000000000,
128.0000000000, 128.0000000000, 128.0000000000, 129.0000000000, 129.0000000000, 129.0000000000,
129.0000000000, 129.0000000000, 129.0000000000, 130.0000000000, 130.0000000000, 130.0000000000,
130.0000000000, 130.0000000000, 130.0000000000, 130.0000000000, 131.0000000000, 131.0000000000,
131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000, 131.0000000000,
132.0000000000, 132.0000000000, 132.0000000000, 132.0000000000, 133.0000000000, 133.0000000000,
133.0000000000, 133.0000000000, 133.0000000000, 134.0000000000, 134.0000000000, 134.0000000000,
134.0000000000, 134.0000000000, 135.0000000000, 135.0000000000, 135.0000000000, 135.0000000000,
135.0000000000, 135.0000000000, 136.0000000000, 136.0000000000, 137.0000000000, 137.0000000000,
137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000, 137.0000000000,
138.0000000000, 138.0000000000, 138.0000000000, 138.0000000000, 138.0000000000, 139.0000000000,
139.0000000000, 139.0000000000, 140.0000000000, 140.0000000000, 140.0000000000, 140.0000000000,
140.0000000000, 140.0000000000, 141.0000000000, 141.0000000000, 141.0000000000, 141.0000000000,
142.0000000000, 142.0000000000, 143.0000000000, 143.0000000000, 143.0000000000, 143.0000000000,
144.0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 144.0000000000, 145.0000000000,
145.0000000000, 145.0000000000, 146.0000000000, 146.0000000000, 146.0000000000, 146.0000000000,
146.0000000000, 146.0000000000, 147.0000000000, 147.0000000000, 147.0000000000, 147.0000000000,
147.0000000000, 148.0000000000, 148.0000000000, 149.0000000000, 149.0000000000, 149.0000000000,
149.0000000000, 149.0000000000, 149.0000000000, 149.0000000000, 150.0000000000, 150.0000000000,
150.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 150.0000000000, 151.0000000000,
151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000, 151.0000000000,
152.0000000000, 152.0000000000, 152.0000000000, 153.0000000000, 153.0000000000, 153.0000000000,
153.0000000000, 153.0000000000, 153.0000000000, 154.0000000000, 154.0000000000, 154.0000000000,
154.0000000000, 154.0000000000, 154.0000000000, 155.0000000000, 155.0000000000, 155.0000000000,
155.0000000000, 155.0000000000, 156.0000000000, 156.0000000000, 156.0000000000, 156.0000000000,
156.0000000000, 156.0000000000, 157.0000000000, 157.0000000000, 157.0000000000, 157.0000000000,
158.0000000000, 158.0000000000, 158.0000000000, 158.0000000000, 158.0000000000, 159.0000000000,
159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 159.0000000000, 160.0000000000,
160.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 160.0000000000, 161.0000000000,
161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 161.0000000000, 162.0000000000,
162.0000000000, 162.0000000000, 162.0000000000, 162.0000000000, 162.0000000000, 163.0000000000,
163.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000, 164.0000000000,
164.0000000000, 164.0000000000, 165.0000000000, 165.0000000000, 165.0000000000, 165.0000000000,
165.0000000000, 165.0000000000, 166.0000000000, 166.0000000000, 166.0000000000, 166.0000000000,
```

```
166.0000000000, 166.0000000000, 166.0000000000, 167.0000000000, 167.0000000000, 167.0000000000,
167.0000000000, 167.0000000000, 167.0000000000, 168.0000000000, 168.0000000000, 168.0000000000,
168.0000000000, 168.0000000000, 168.0000000000, 168.0000000000, 169.0000000000, 169.0000000000,
169.0000000000, 169.0000000000, 169.0000000000, 169.0000000000, 170.0000000000, 170.0000000000,
170.0000000000, 170.0000000000, 170.0000000000, 170.0000000000, 171.0000000000, 171.0000000000,
171.0000000000, 171.0000000000, 171.0000000000, 171.0000000000, 172.0000000000, 172.0000000000,
172.0000000000, 172.0000000000, 172.0000000000, 173.0000000000, 173.0000000000, 173.0000000000,
173.0000000000, 173.0000000000, 173.0000000000, 174.0000000000, 174.0000000000, 174.0000000000,
174.0000000000, 174.0000000000, 174.0000000000, 175.0000000000, 175.0000000000, 175.0000000000,
175.0000000000, 175.0000000000, 175.0000000000, 175.0000000000, 176.0000000000, 176.0000000000,
176.0000000000, 176.0000000000, 176.0000000000, 177.0000000000, 177.0000000000, 177.0000000000,
177.0000000000, 177.0000000000, 177.0000000000, 178.0000000000, 178.0000000000, 178.0000000000,
178.0000000000, 178.0000000000, 179.0000000000, 179.0000000000, 179.0000000000, 179.0000000000,
179.0000000000, 179.0000000000, 180.0000000000, 180.0000000000, 180.0000000000, 180.0000000000,
181.0000000000, 181.0000000000, 181.0000000000, 181.0000000000, 181.0000000000, 182.0000000000,
182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000, 182.0000000000,
183.0000000000, 183.0000000000, 183.0000000000, 183.0000000000, 183.0000000000, 183.0000000000,
184.0000000000, 184.0000000000, 184.0000000000, 184.0000000000, 184.0000000000, 184.0000000000,
185.0000000000, 185.0000000000, 185.0000000000, 185.0000000000, 185.0000000000, 185.0000000000,
186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000, 186.0000000000,
187.0000000000, 187.0000000000, 187.0000000000, 187.0000000000, 187.0000000000, 188.0000000000,
188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000, 188.0000000000,
189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000, 189.0000000000,
190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000, 190.0000000000,
190.0000000000, 190.0000000000, 191.0000000000, 191.0000000000, 191.0000000000, 191.0000000000,
191.0000000000, 191.0000000000, 191.0000000000, 192.0000000000, 192.0000000000, 192.0000000000,
192.0000000000, 192.0000000000, 192.0000000000, 192.0000000000, 193.0000000000, 193.0000000000,
193.0000000000, 193.0000000000, 194.0000000000, 194.0000000000, 194.0000000000, 194.0000000000,
194.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000, 195.0000000000,
195.0000000000, 196.0000000000, 196.0000000000, 196.0000000000, 196.0000000000, 197.0000000000,
197.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 197.0000000000, 198.0000000000,
198.0000000000, 198.0000000000, 199.0000000000, 199.0000000000, 199.0000000000, 199.0000000000,
199.0000000000, 199.0000000000, 200.0000000000, 200.0000000000, 200.0000000000, 200.0000000000,
200.0000000000, 200.0000000000, 201.0000000000, 201.0000000000, 201.0000000000, 201.0000000000,
201.0000000000, 201.0000000000, 201.0000000000, 202.0000000000, 202.0000000000, 203.0000000000,
203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000, 203.0000000000,
204.0000000000, 204.0000000000, 204.0000000000, 204.0000000000, 204.0000000000, 205.0000000000,
205.0000000000, 205.0000000000, 206.0000000000, 206.0000000000, 206.0000000000, 206.0000000000,
206.0000000000, 206.0000000000, 207.0000000000, 207.0000000000, 207.0000000000, 207.0000000000,
208.0000000000, 208.0000000000, 209.0000000000, 209.0000000000, 209.0000000000, 210.0000000000,
210.0000000000, 210.0000000000, 210.0000000000, 210.0000000000, 210.0000000000, 211.0000000000,
211.0000000000, 211.0000000000, 211.0000000000, 212.0000000000, 212.0000000000, 212.0000000000,
212.0000000000, 212.0000000000, 213.0000000000, 213.0000000000, 213.0000000000, 213.0000000000,
213.0000000000, 213.0000000000, 214.0000000000, 214.0000000000, 214.0000000000, 214.0000000000,
215.0000000000, 215.0000000000, 215.0000000000, 215.0000000000, 215.0000000000, 216.0000000000,
216.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 217.0000000000, 217.0000000000,
217.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000, 218.0000000000,
218.0000000000, 219.0000000000, 219.0000000000, 219.0000000000, 219.0000000000, 219.0000000000,
219.0000000000, 220.0000000000, 220.0000000000, 220.0000000000, 221.0000000000, 221.0000000000,
221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 221.0000000000, 222.0000000000,
222.0000000000, 222.0000000000, 222.0000000000, 222.0000000000, 223.0000000000, 223.0000000000,
223.0000000000, 223.0000000000, 223.0000000000, 223.0000000000, 224.0000000000, 224.0000000000,
224.0000000000, 224.0000000000, 224.0000000000, 224.0000000000, 225.0000000000, 225.0000000000,
225.0000000000, 225.0000000000, 225.0000000000, 225.0000000000, 226.0000000000, 226.0000000000,
226.0000000000, 226.0000000000, 226.0000000000, 226.0000000000, 227.0000000000, 227.0000000000,
227.0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 227.0000000000, 228.0000000000,
228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000, 228.0000000000,
229.0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 229.0000000000, 230.0000000000,
230.0000000000, 230.0000000000, 230.0000000000, 230.0000000000, 231.0000000000, 231.0000000000,
231.0000000000, 231.0000000000, 231.0000000000, 232.0000000000, 232.0000000000, 232.0000000000,
232.0000000000, 232.0000000000, 233.0000000000, 233.0000000000, 233.0000000000, 233.0000000000,
233.0000000000, 233.0000000000, 233.0000000000, 234.0000000000, 234.0000000000, 234.0000000000,
234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 234.0000000000, 235.0000000000,
235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 235.0000000000, 236.0000000000,
236.0000000000, 236.0000000000, 236.0000000000, 236.0000000000, 236.0000000000, 237.0000000000,
237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 237.0000000000, 238.0000000000,
238.0000000000, 238.0000000000, 238.0000000000, 238.0000000000, 239.0000000000, 239.0000000000,
239.0000000000, 239.0000000000, 239.0000000000, 239.0000000000, 240.0000000000, 240.0000000000,
240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000, 240.0000000000,
241.0000000000, 241.0000000000, 241.0000000000, 241.0000000000, 241.0000000000, 241.0000000000,
242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000, 242.0000000000,
242.0000000000, 242.0000000000, 243.0000000000, 243.0000000000, 243.0000000000, 243.0000000000,
243.0000000000, 243.0000000000, 244.0000000000, 244.0000000000, 244.0000000000, 244.0000000000,
244.0000000000, 245.0000000000, 245.0000000000, 245.0000000000, 245.0000000000, 245.0000000000,
246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 246.0000000000, 247.0000000000,
247.0000000000, 247.0000000000, 247.0000000000, 248.0000000000, 248.0000000000, 248.0000000000,
248.0000000000, 248.0000000000, 249.0000000000, 249.0000000000, 249.0000000000, 249.0000000000,
249.0000000000, 249.0000000000, 250.0000000000, 250.0000000000, 250.0000000000, 250.0000000000,
250.0000000000, 250.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000,
251.0000000000, 251.0000000000, 251.0000000000, 251.0000000000, 252.0000000000, 252.0000000000,
252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 252.0000000000, 253.0000000000,
253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000, 253.0000000000,
254.0000000000, 254.0000000000, 254.0000000000, 254.0000000000, 254.0000000000, 255.0000000000,
255.0000000000, 255.0000000000, 256.0000000000, 256.0000000000, 256.0000000000, 256.0000000000,
257.0000000000, 257.0000000000, 257.0000000000, 257.0000000000, 257.0000000000, 257.0000000000,
258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000, 258.0000000000,
258.0000000000, 259.0000000000, 259.0000000000, 259.0000000000, 259.0000000000, 260.0000000000,
```

```
       260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 260.0000000000, 261.0000000000,
       261.0000000000, 261.0000000000, 262.0000000000, 262.0000000000, 263.0000000000, 263.0000000000,
       263.0000000000, 263.0000000000]
00051
00052 calcWeightsGaussCorrect = [[-0.57735026918962573105886800,
       0.57735026918962573105886800],[1.00000000000000004440892099, 1.00000000000000004440892099]]
00053
00054 gaußLegendreCorrect = [[0.21132486540518713, 0.21132486540518713, 0.7886751345948129,
       0.7886751345948129], [0.16666666666666669, 0.6220084679281462, 0.044658198738520456,
       0.16666666666666669], [0.19716878364870338, 0.19716878364870338, 0.052831216351296825,
       0.052831216351296825]]
00055
00056 reducedSFCorrect = [-0.27937980331459932870075136, -0.33966732733873783445233130,
       158.30535989173870348167838529, 111.95922396290268352458951995, 58.109038590106571575688857656,
       193.39790162602255918500304688, 268.25033740186444219943950884, 189.82215394456304125014867168,
       0.25883372957144956671982072, 98.748706334028099718125304 21, 215.53653410699894266144838184,
       327.39158447388479089568136260, 284.96744613445838467669091187, 202.31101004349304162133194041,
       0.82513685120877511280923500, 105.22484885820357192187657347, 223.08256339289710012963041663,
       364.76793579171919645887101069, 347.61723438903078431394533254, 186.13386039326010745753592346,
       132.63331666423582078095932957, 0.93045442511324694123686641, 69.59377646537200234888587147,
       377.49981102452846926098573022, 387.86131916037487599169253372, 228.06934028761415333974582609,
       -0.39282069172599420880942489, 0.28340911989108974466944346, 1.32922047073893168089853134,
       0.77036064878831922442259383, 402.01980094098689733073115349, 253.64580857732585172925610095,
       -0.71137220262248490598722128, -187.02281867595976905249699485, -132.12396908339931655973487068,
       1.00935338967148879874002887, -68.31938968731270733769633807, 262.84597566425361492292722687,
       -1.01378116441184573659484158, -229.49291759567395843077974860, -285.00916079306904293844127096,
       -202.15395986486728929776290897, 0.20369749822588545384860481, -105.12807714412967641237628413,
       -1.65128343645850605980740511, -255.65451390973146317264763638, -348.80806087323372821629163809,
       -268.84160316836425863584736362, -190.14226619109740568092092872, -0.35372141836276682624884415,
       -98.67697413471481127089646179, -265.72733211685635978938080370, -390.53074044891235416798735969,
       -328.54385597348704095566063188, -159.10063256399396891538344789, -112.66830824900223717577318894,
       -0.48136222438574449711268244, -58.49296577544600950204767287, -406.26733909792579879649565555,
       -367.00125201198864033358404413, -194.68519835904922388181148563, -0.38739421188725309264100360,
       -380.72973951162691719218855724, -216.95201141250419141215388663, -224.64681969190800714386568870,
       -57.64885629013803480802380363, -111.76778906363414023417135468, -97.23151838524054824119957630,
       -158.57787734712869109898747411, -189.13083157351150020986096933, -103.47360646666479055966192391,
       -194.37812613590944010866223834, -268.14533020187656120469910093, -201.00144759920507908645959105,
       -67.33160591182809184829238802, -216.97908089605186887638410553, -328.52149241710583282838342711,
       -284.82532200213120177068049088, -130.61038743255170402335352264, 1.1740831305907946102706737,
       -225.01886593214194931533711497, -366.28282561516010673585697077, -347.97810273027931771139265038,
       -184.84955361689296182703401428, 1.35575884552714853725774447, 68.83865863464234280399978161,
       -379.59834765834438030651654117, -389.38131238244437781759188510, -226.65542721376317558679147623,
       2.45142690491787362816467066, 132.61870525252811603422742337, 104.51756439557146904917317443,
       -402.65102785189810674637556076, -252.04129957093991265537624713, 2.66640808738427992352626461,
       188.46031263508777442439168226, 202.68330394428298291131795850, 98.049191606125660314319247 85,
       -262.11224472208607494394527748, 2.80043348997752872264754842, 231.30365585971119912755966652,
       286.80511684477886547028901987, 190.23232474250315249264531303, 57.84805143771897917304158909,
       2.85708460601647828980276245, 258.69203442119561486833845265, 350.58867061999404768357635476,
       268.52259877181154479330871254, 112.10337467749660333993233507, 266.08331708857917874411214143,
       390.90976985716883973509538919, 329.01088679331462572008604184, 158.71128445003517981604090892,
       404.59245903407168043486308306, 366.35434029559155533206649125, 194.19573855113847571374208201,
       379.94898461640434561559231952, 216.17590064203761812677839771, 223.85736931998752652361872606,
       215.30222433132459514126821887, 193.26277766072317376710998360, 364.27701325326768255763454363,
       157.72987052894848147843731567, 326.94128157610799689791747369, 388.09280021282381767377955839,
       111.18402577739000491874321597, 266.54277820804679777211276814, 346.73233678344666941484319977,
       252.57711524406573744272463955, 57.016391657150371941042976669, 187.85218553479762704228051007,
       282.67168803322431358537869528, 224.67611712193377115909243003, -3.14326924385573391873549554,
       -0.90010570311670168308637585, 96.17069437543644028210110264, 199.33582061541360985756909940,
       182.27846100397448481089668348, -4.57611186272201919678082049, -257.99057047583272606061655097,
       -1.82557398572480522602745623, 101.69189406672646214246924501, 127.54445573081555664884945145,
       -4.49732115287934419711746159, -232.30379071310053973320464138, -392.35336998126877006143331528,
       -2.65962064013652721428115910, 64.23306507635152229340746999, -4.17823235146674676343536703,
       -190.62136057342786443769000471, -350.97981405658060793939512223, -367.33195935840376478154212236,
       -3.92525467655692494872710085, -4.21737374407493348371644970, -136.02000312436450712993973866,
       -288.23032697439572302755550481, -329.76997539886201593617442995, -217.04860831373792962040170096,
       -4.22506971989652058141473390, -72.29251053978725849447073415, -205.62211432499287866448867135,
       -270.03888658781465892388951033, -195.10425396920604157458001282, -3.75470502920668458557429403,
       -107.71214521643315720211830921, -191.64672875892048864443495404, -159.55913514871488700919144321,
       -2.82958247867568246647351771, -100.33839131188909732372849248, -113.28188975877471023068210343,
       -2.02984471265764909730933141, -59.20137760431371987124293810, -1.13006153433920530915202107,
       -216.94986764694439784761925694, -194.17815277072924118328955956, -366.46414265432213142048567533,
       -58.68817436396830089506693184, -158.58371851794368012633640319, -328.49641010394003615147084929,
       -389.37906114078765540398308076, -112.59894550018677250591281336, -99.47273320793448192489449866,
       -268.50544005644428580126259476, -349.78532743252520731402910315, -254.30643984757679731956159230,
       -190.57684701519576719874748960, -106.72241271551203567469201516, -286.77766766087427185993874446,
       -228.97129601689493938465602696, 0.65888959003758174048925866, -203.43166351223138121895317454,
       -71.95880175563581815367797390, -188.94812353702479867934016511, -0.54028097490086324228286685,
       255.71015351091924117099551950, -135.09478864364308492440613918, -4.18373247132461756336851977,
       -1.97138030481194448384485440, 228.04006865312763352449110243, 391.09379142082968883187277243,
       -3.47586424690169337736733723, 65.01418211509854927498963661, 184.65606105019628557784017175,
       348.36366144437067760009085760, 366.20261451786950601672288030, 128.93974257358527795615542386,
       101.83529783301392512839811388, 283.51551025120562599113327451, 327.60501736958264018539921381,
       216.03661549146309539537469391, 200.00439085138680184172699228, 96.43013782801264710542454850,
       267.22322299296490655251545832, 193.49255409229277802296564914, 188.08418865096339800402347464,
       57.04436608707574407617357792, 157.96464661316298361271037720, 111.31922987220724508006242104]
00057
00058 SFCorrectValue = [-0.27937980331459933, 158.30535989173870348, -0.27937980331459933,
```

```
      111.95922396290268352, 58.10903859010657158, -0.27937980331459933, 193.39790162602255919,
      -0.27937980331459933, 268.25033740186444220, 189.82215394056304125, 0.25883372957144957,
      -0.27937980331459933, 98.74870633402809972, -0.27937980331459933, 215.53653410699894266,
      327.39158447388479090, 284.96744613445838468, 202.31101004349304162, 0.82513685120877511,
      105.22484885820357192, -0.27937980331459933, 223.08256339289710013, 364.76793579171919646,
      347.61723438903078431, 186.13386039326010746, 132.63331666423582078, 0.93045442511324694,
      69.59377646537200235, 377.49981102452846926, 387.86131916037487599, 228.06934028761415334,
      -0.39282069172599421, 0.28340911989108974, 1.32922047073893168, 0.77036064878831922,
      402.01980094098689733, 253.64580857732585173, -0.71137220262248491, -187.02281867595976905,
      -132.12396908339931656, 1.00935338967148880, -68.31938968731270734, 262.84597566425361492,
      -1.01378116441184574, -229.49291759567395843, -285.00916079306904294, -202.15395986486728930,
      0.20369749822588545, -105.12807714412967641, -1.65128343645850606, -255.65451390973146317,
      -348.80806087323372822, -268.84160316836425864, -190.14226619109740568, -0.35372141836276683,
      -98.67697413471481127, -265.72733211685635979, -390.53074044891235417, -328.54385597348704096,
      -159.10063256399396892, -112.66830824900223718, -0.481362224385744 50, -58.49296577544600950,
      -406.26733909792579880, -367.00125201198864033, -194.68519835904922388, -0.33966732733873783,
      -0.33966732733873783, -0.38739421188725309, -0.33966732733873783, -380.72973951162691719,
      -216.95201141250419141, -0.33966732733873783, -0.33966732733873783, -224.64681969190800714,
      -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, -57.64885629013803481,
      -0.27937980331459933, -111.76778906363414023, -97.23151838524054824, -158.57787734712869110,
      -0.27937980331459933, -189.13083157351150021, -103.47360646666479056, -0.27937980331459933,
      -194.37812613590944011, -268.14533020187656120, -201.00144759920507909, -67.33160591182809185,
      -216.97908089605186888, -0.27937980331459933, -328.52149241710583283, -284.82532200213120177,
      -130.61038743255170402, 1.17408313059079461, -225.01886593214194932, -0.27937980331459933,
      -366.28282561516010674, -347.97810273027931771, -184.84955361689296183, 1.35575884552714854,
      68.83865863464234280, -379.59834765834438031, -389.38131238244437782, -226.65542721376317559,
      2.45142690491787363, 132.61870525252811603, 104.51756439557146905, -402.65102785189810675,
      -252.04129957093991266, 2.66640808738427992, 188.46031263508777442, 202.68330394428298291,
      98.04919160612566031, -262.11224472208607494, 2.80043348997752872, 231.30365585971119913,
      286.80511684477886547, 190.23232474250315249, 57.84805143771897917, 2.85708460601647829,
      258.69203442119561487, 350.58867061999404768, 268.52259877181154479, 112.10337467749660334,
      -0.33966732733873783, 266.08331708857917874, 390.90976985716883974, 329.01088679331462572,
      158.71128445003517982, -0.33966732733873783, 404.59245903407168043, 366.35434029559155533,
      194.19573855113847571, -0.33966732733873783, 379.94898461640434562, 216.17590064203761813,
      -0.33966732733873783, 223.85736931998752652, -0.33966732733873783, -0.33966732733873783,
      -0.27937980331459933, 215.30222433132459514, -0.27937980331459933, 193.26277766072317377,
      364.27701325326768256, -0.27937980331459933, 157.72987052894848148, 326.94128157610799690,
      388.09280021282381767, -0.27937980331459933, 111.18402577739000492, 266.54277820804679777,
      346.73233678344666941, 252.57711524406573744, -0.27937980331459933, 57.01639165715037194,
      187.85218553479762704, 282.67168803322431359, 224.67611712193377116, -3.143269243855733 92,
      -0.27937980331459933, -0.90010570311670168, 96.17069437543644028, 199.33582061541360986,
      182.27846100397448481, -4.57611186272201920, -257.99057047583272606, -1.82557398572480523,
      101.69189406672646214, 127.54445573081555665, -4.49732115287934420, -232.30379071310053973,
      -392.35336998126877006, -2.65962064013652721, 64.23306507635152229, -4.17823235146674676,
      -190.62136057342786444, -350.97981405658060794, -367.33195935840376478, -3.92525467655692495,
      -4.21737374407493348, -136.02000312436450713, -288.23032697439572303, -329.76997539886201594,
      -217.04860831373792962, -4.22506971989652058, -72.29251053978725849, -205.62211432499287866,
      -270.03888658781465892, -195.10425396920604157, -0.33966732733873783, -3.75470502920668459,
      -107.71214521643315720, -191.64672875892048864, -159.55913514871488701, -0.33966732733873783,
      -2.82958247867568247, -100.33839131188909732, -113.28188975877471023, -0.33966732733873783,
      -2.02984471265764910, -59.20137760431371987, -0.33966732733873783, -1.13006153433920531,
      -0.33966732733873783, -0.33966732733873783, -0.27937980331459933, -216.94986764694439785,
      -0.27937980331459933, -194.17815277072924118, -366.46414265432213142, -0.27937980331459933,
      -58.68817436396830090, -0.27937980331459933, -158.58371851794368013, -328.49641010394003615,
      -389.37906114078765540, -0.27937980331459933, -112.59894550018677251, -99.47273320793448192,
      -268.50544005644428580, -349.78532743252520731, -254.30643984757679732, -190.57684701595376720,
      -106.72241271551203567, -286.77766766087427186, -228.97129601689493938, 0.65888959003758174,
      -203.43166351223138122, -71.95880175563581815, -188.94812353702479868, -0.54028097490086324,
      255.71015351091924117, -135.09478864364308492, -4.18373247132461756, -1.97138030481194448,
      228.04006865312763352, 391.09379142082968883, -3.47586424690169338, 65.01418211509854927,
      184.65606105019628558, 348.36366144437067760, 366.20261451786950602, 128.93974257358527780,
      101.83529783301392513, 283.51551025120562599, 327.60501736958264019, 216.03661549146309540,
      200.00439085138680184, 96.43013782801264711, 267.22322299296490655, 193.49255409229277802,
      -0.33966732733873783, 188.08418865096339800, 57.04436608707574408, 157.96464661316298361,
      -0.33966732733873783, 111.31922987220724508, -0.33966732733873783, -0.33966732733873783,
      -0.33966732733873783]
00059
00060 bFieldCorrectValue = [[-0.00000466916987661623140515657, -0.00000469774735703660250055961,
      -0.00000395939995441359537849948, -0.00000482221010938081362973313, -0.00000396550195293026650080566,
      -0.00000387749965995507482504874, -0.00000382771914822608855781148, -0.00000292862967710368168300470,
      -0.00000309944588351366663844040, -0.00000326900335531990481345490, -0.00000454936400457388206747429,
      -0.00000558752101129238260083512, -0.00000408059138231045771583226, -0.00000355106400521909434142326,
      -0.00000435087021876521834478800, -0.00000601269263477531150000705, -0.00000422084203782920821688886,
      -0.00000318190138831836488695379, -0.00000410009973941543825440466, -0.00000596980094276740785646726,
      -0.00000437558746190408254440234, -0.00000290454751947236613300574, -0.00000382714510184938365401023,
      -0.00000548811576122437301257333, -0.00000452500225311201280219012, -0.00000289673824026572774027722,
      -0.00000355797907710025893013046, -0.00000472917096365148288520958, -0.00000464202966810378363380275,
      -0.00000325927840943272435566430, -0.00000331142973766262746783385, -0.00000391818849209657636467391,
      -0.00000209963437544594680927643, -0.00000334845606049689288255403, -0.00000442231665001654005299307,
      -0.00000340778215041781557426582, -0.00000496401499704663627101850, -0.00000371020589188032213470627,
      -0.00000647174234685046458954343, -0.00000310184058961244829062285, -0.00000741610960273363378910619,
      -0.00000230211586279070535689746, -0.00000754218056388060709003445, -0.00000166104789262922323326264,
      -0.00000688055635928999741163627, -0.00000152421125169085425935086, -0.00000570765164247120032554915,
      -0.00000834218389464975617795476, -0.00000857509678229675104794115, -0.00000171698141817773268018820,
      -0.00000083362705561108169433266, -0.00000779149266707861698196078, -0.00000061122225394470813803995,
      -0.00000634459329939409620016319, -0.00000132985936788230726855145, -0.00000474768270446881180025947,
      -0.00000294559945549423868964008, -0.00000349515548030209205720683, -0.00000506152148848857683034784,
```

```
-0.00000363192483452629967890547, -0.00000705892632586430391890102, -0.00000280316637579680072320460,
-0.00000484372545126640032442291, -0.00000351759447383940304743635, -0.00000278445555434836104581934,
-0.00000506790710227895837522742, -0.00000360543134416433626724991, -0.00000723002305488891725015511,
-0.00000270154459553407161272231, -0.00000862725220637769032246307, -0.00000151603425631581637699668,
-0.00000889681845434635380212227, -0.00000054574943740535989470102, -0.00000807459544423345471292688,
-0.00000028768415648587549632195, -0.00000653979801767764028344381, -0.00000104852021517869108351137,
-0.00000172984723925869242351476, -0.00000083983432612446153178254, -0.00000843900811589068215474302,
-0.00000767582118307908633147649, -0.00000059726856794431001312451, -0.00000625821688852544495012837,
-0.00000128468751960543469050485, -0.00000469346301964793005857693, -0.00000286515300439714568582678,
-0.00000347076187887664550473497, -0.00000494941956996801184741341, -0.00000363493590034705472318136,
-0.00000692469606038360307309750, -0.00000281267864454127884416695, -0.00000819936244305945920781559,
-0.00000317196252454694048298657, -0.00000471374717555926789889532, -0.00000371565160363171774840236,
-0.00000336022805812820178159180, -0.00000617358952479633905262779, -0.00000311900978890395319081648,
-0.00000710683808874491928337606, -0.00000232528792457305233048830, -0.00000725756731943215569298955,
-0.00000167147731172188549259223, -0.00000664643984443812749413928, -0.00000149442711094559785073212,
-0.00000553693025919358604780215, -0.00000200332775657395782884754, -0.00000431645828795938008899278,
-0.00000552943336618137946936440, -0.00000513784876522024205164861, -0.00000291349221850059231320802,
-0.00000284655847889421133734954, -0.00000448044829528352128109872, -0.00000310574162993285972093525,
-0.00000376609468348992355569939, -0.00000367413552144735505288614, -0.00000320068755163319901570185,
-0.00000441266274488435117693110, -0.00000383452939051735118357973, -0.00000510130172372361619949864,
-0.00000357225423395365277440933, -0.00000551868766810142620429357, -0.00000320964682172228580624477,
-0.00000301378877941614634153886, -0.00000410778685826763726801842, -0.00000388664939509098225812892,
-0.00000397227444580496633736669, -0.00000410069159332783634284211, -0.00000368643251657461040284966,
-0.00000424376923364454084695299, -0.00000352000049615883929109844, -0.00000437117894384989181643785,
-0.00000337662784325707268512644, -0.00000444378070030683556407292, -0.00000324438745338572602120797,
-0.00000442575475476003422144623, -0.00000312146444177834044445626, -0.00000430506140600200794699842,
-0.00000162539700129068705292232, -0.00000201042117077484983964498, -0.00000602517698315944672481353,
-0.00000574081659120288884194097, -0.00000247517126240858007990759, -0.00000494816587723671617155257,
-0.00000282491005183331607728903, -0.00000383106878716190527245583, -0.00000410633847534577906126596,
-0.00000270729373226552707089668, -0.00000462023025166119135509089, -0.00000188570036238920448643473,
-0.00000526327669295933915524733, -0.00000153163976212105700181127, -0.00000581066380738314988772737,
-0.00000421442728380309715322319, -0.00000504756703827697536253027, -0.00000170989697017502109315835,
-0.00000036820565124675968927425, -0.00000610496487403976930850771, 0.00000014797071539980571240752,
-0.00000699815409738582605510314, -0.00000013954325645392500584017, -0.00000732531126871401128082322,
-0.00000095744908611779592605114, -0.00000681484750388092570876032, -0.00000191909864520732659405339,
-0.00000546253804236683169326784, -0.00000265938288856101964844876, -0.00000358298816469999955667126,
-0.00000812796083866744625931900, -0.00000745753776634870499140937, -0.00000023423816798760116821121,
-0.00000153049770289275655546350, -0.00000573509010013384775339071, -0.00000253874852852054134442214,
-0.00000336247187884871456792875, -0.00000427797885784376955985338, -0.00000100916025111619696138707,
-0.00000531326814108958386258892, 0.00000066596593324540348829042, -0.00000663341610125664479021435,
0.00000128644991395470098325792, -0.00000774121282552941076843690, 0.00000087139595040765255933263,
-0.00000068697680683621176608519, 0.00000108710647006848568885107, -0.00000537257430265921628919149,
-0.00000676238790144111092485007, 0.00000173567337224714148292688, -0.00000792044380864004106987224,
0.00000127289596233354430259224, -0.00000830491374041144473706118, 0.00000006226699218459259021648,
-0.00000756441053941680220911847, -0.00000136057344123341247680604, -0.00000571394070933897222481189,
-0.00000247762055243248720509241, -0.00000318552582204159535531713, -0.00000428507087423901900883217,
-0.00000009976191065762934721919, -0.00000142722840623724671940123, -0.00000712189078862439074952065,
-0.00000540387404400172997886737, -0.00000248162448249075073513346, -0.00000307367481937223088330293,
-0.00000423231060258485362966102, -0.00000077719992525453455582050, -0.00000521221734472884650916707,
0.00000085326870520938784575214, -0.00000646726797645818615197085, 0.00000145036643128491965586667,
-0.00000750493729109076359180255, 0.00000102295780067718125160708, -0.00000782893957445994090273058,
-0.00000485233344675319492553820, -0.00000578757375039132682651875, 0.00000000202870802739352439407,
0.00000049412370812489573363346, -0.00000655251389230868966172651, 0.00000017652638644477472342377,
-0.00000676767675254085845519330, -0.00000068355491654192516321755, -0.00000619242409243209346216951,
-0.00000171291595419151859865624, -0.00000484496278180954133459057, -0.00000254671182411419128217844,
-0.00000303232682677181289733199, -0.00000412542168294146950938660, -0.00000124926818805093130415769,
-0.00000490005523125601636042024, -0.00000410079205118944214552819, -0.00000216780225761254019133280,
-0.00000265993509375987947243136, -0.00000304451184640827180187712, -0.00000397844617824152245097718,
-0.00000200886521446264364309923, -0.00000434337196074779258587713, -0.00000126497306347134013685340,
-0.00000482034513729443117011119, -0.00000097047708767977640531154, -0.00000519910820894511092936461,
-0.00000112395492016971269517501, -0.00000526988114269102274377695, -0.00000159029777910767285258053,
-0.00000279007641965746346659734, -0.00000270792053478371691268840, -0.00000370586139530073586841876,
-0.00000363983103204849961162266, -0.00000267094054963526342743154, -0.00000354731158106263330496858,
-0.00000267469876821535449356150, -0.00000341971438965730736118400, -0.00000271767379212010974846143,
-0.00000325926598218750463730495, -0.00000280154957479585715502432, -0.00000308313457501994204050283,
-0.00000381168789036198818887934, -0.00000291868215581486715136644, -0.00000375857466944896834637382,
-0.00000327537143583890508134987, -0.00000294811285364807816865860, -0.00000243131872314751138567930,
-0.00000312667996150283317564845, -0.00000364870836547505383420432, -0.00000382781652651242955328345,
-0.00000318312915541018931139816, -0.00000431394405874985608479420, -0.00000260736864294356355278804,
-0.00000444729099825922800318795, -0.00000210229289719470295108865, -0.00000422375510895062963596502,
-0.00000184875947709459478784043, -0.00000376721906150404020553919, -0.00000196045186722052175047398,
-0.00000592342084085147574894345, -0.00000554476070792437015041136, -0.00000081400334724811556570191,
-0.00000042532442268500512611083, -0.00000469959170441616449028681, -0.00000073734569966215719075771,
-0.00000375403550559192771064504, -0.00000173768490097683139598264, -0.00000307610972027205064872847,
-0.00000316560220853134947159428, -0.00000351282516468181480053619, -0.00000459692139732042535471383,
-0.00000270152777123880139044138, -0.00000560450046418568346008431, -0.00000168719999321485117244079,
-0.00000342363473793270677307705, -0.00000316555516410657517936450, -0.00000319946793265786186096140,
-0.00000511476964655244583317752, -0.00000238495263015079001238297, -0.00000647013587860897266575274,
-0.00000108257378739797452410357, -0.00000691468568123992726007524, 0.00000003167064363558358119560,
-0.00000643415451976168424738298, 0.00000050771520929822222940154, -0.00000532987843263442524397409,
0.00000006241015339472573124498, -0.00000408020987444468504336086, -0.00000128761938519727446140081,
0.00000030530096339922681130937, 0.00000080493163372144399727714, -0.00000675910653127617878010644,
-0.00000556357340243448327514334, 0.00000031053661946659168990848, -0.00000420489142170216773552929,
-0.00000115830494527970682990653, -0.00000320303497704641381598530, -0.00000323198101491799745914346,
-0.00000339408033863997476140863, -0.00000053100100272048253105408, -0.00000228056363624846953358657,
-0.00000678469324238176057665977, -0.00000088477655209753822453925, -0.00000727413322656034234823174,
-0.00000326897805431183154865329, -0.00000516167892761611436195247, -0.00000342850679923833693513798,
```

-0.00000240432545669479192157665, -0.00000650681217169613519598617, -0.00000112505414714324030004159,
-0.00000695190053809194577689231, -0.00000003810938074688627256705, -0.00000647383878269336087677051,
0.00000041456647116172939637499, -0.00000536782825442866377065607, -0.00000004038698461162372195087,
-0.00000411090341017683102498633, -0.00000138073205653745637771015, -0.00000318372807349149021831011,
-0.00000562727528744765789656080, -0.00000477658189218249504330539, -0.00000061227845890646505921961,
-0.00000094827088884762486339924, -0.00000038154784808717637708963, -0.00000193443873572884990087164,
-0.00000311210636208381668708044, -0.00000331869797143930478063121, -0.00000352199795198601431594869,
-0.00000470494659767096951681273, -0.00000273849507456990307887549, -0.00000568889862162176860514674,
-0.00000176952899103109435089657, -0.00000600451785619847786006813, -0.00000095149715620158576508038,
-0.00000366110089627885071218260, -0.00000032341855918766746785960, -0.00000402600863687437650643638,
-0.00000446949584777140901457708, -0.00000272400146837823654553967, -0.00000458672159083538902730462,
-0.00000230235148654503241273011, -0.00000435593059868150186562854, -0.00000212883305276795807244646,
-0.00000388553992036171978014490, -0.00000228752806541513610690507, -0.00000336762096545540925805881,
-0.00000275052968444129470779961, -0.00000300120531727028257365920, -0.00000339065697054311368337427,
-0.00000391432606865399758898881, -0.00000386718982324291977298574, -0.00000283760083917508459307069,
-0.00000284060446008821598426735, -0.00000372019916778028846761242, -0.00000287057319934658170746864,
-0.00000349091419644113533195316, -0.00000382583235164472675650946, -0.00000324501647033728225547034,
-0.00000381864180355396068828862, -0.00000304896794445573561912121, -0.00000384790812524552411816358,
-0.00000292626512526106090971768, -0.00000038923083808633649041687, -0.00000286273819758726964223856,
-0.00000249269458175358870586106, -0.00000165548223720961917416301, -0.00000440615533682643821977894,
-0.00000497534078462630395658971, -0.00000128768330081076083257988, -0.00000548659396172719273965289,
-0.00000138575601129449019273030, -0.00000570622881830273955851545, -0.00000180101518164556902500226,
-0.00000545759896474117850602697, -0.00000232071833293005072820412, -0.00000470744494544863683336069,
-0.00000274320869261157984186732, -0.00000361602065907775451240533, -0.00000399266753379723225584122,
-0.00000094366416971130777890231, -0.00000189395963436627874766454, -0.00000682452670458218092519901,
-0.00000556924458937060300330827, -0.00000264195148276336413701443, -0.00000375379032934690861776812,
-0.00000413806093160064300084770, -0.00000189161388808086560446445, -0.00000491098990586778144832115,
-0.00000051185897216237140715239, -0.00000594009063351439326339937, 0.00000006686969838431272388037,
-0.00000684934766681994114335876, -0.00000016224909706652175270746, -0.00000723950492963986931593042,
0.00000019139181879956180534325, 0.00000091452996700144001899081, -0.00000660102640454561844440221,
-0.00000778004690185671689880292, 0.00000060692497525864937735126, -0.00000828851796919586259664112,
-0.00000040724320752175564500555, -0.00000776851786493057902007429, -0.00000163268775993476249311408,
-0.00000618425225844750323279811, -0.00000258587738875787914879199, -0.00000389496260100784876282895,
-0.00000424196549061700353954471, -0.00000154752422241493602209127, -0.00000526036189220428645633770,
-0.00000158428732695602638935414, -0.00000258725125944710940915764, -0.00000647735646465241152224326,
-0.00000404144750392719362050691, -0.00000429081377341761048419336, -0.00000153856466452788968518931,
-0.00000540547194807550160665149, 0.00000032277242119040693690886, -0.00000686381456598564950491144,
0.00000110565623209020424979579, -0.00000814501678208472028113688, 0.00000078861949552759908613274,
-0.00000870371448507973557073749, -0.00000002851078837805833032662, -0.00000815944672833694236064889,
-0.00000669537711149468838303916, -0.00000789813763548346784327137, 0.00000059111690822940169047717,
0.00000033958897532037827406143, -0.00000843450370123183468468522, -0.00000060731020670712381307903,
-0.00000795474925824088177606606, -0.00000176360881033100408983962, -0.00000642607693187222163444007,
-0.00000264950953861316024627484, -0.00000419727137355453706597883, -0.00000427950624441869564276632,
-0.00000189329076971206080654735, -0.00000532885580003296280786730, -0.00000016218104512392893321314,
-0.00000605647666546202333815807, -0.00000435775685590026293258267, -0.00000421210241176505481092629,
-0.00000276658645189778249273769, -0.00000257663455914345744930628, -0.00000504689462856587593453947,
-0.00000120775748752362999718360, -0.00000612872368050149496745190, -0.00000056728422944763438312402,
-0.00000708742265097004913354441, -0.00000068582099788866186343856, -0.00000753550741748602772706461,
-0.00000133551002728087000193395, -0.00000720199719923842319161842, -0.00000215038968822335375453720,
-0.00000220864124536301706543378, -0.00000214455871493291925708458, -0.00000584699542421314695728662,
-0.00000615773447218937881853483, -0.00000236862926813422162815250, -0.00000603227633872227872430900,
-0.00000269205395604865386263152, -0.00000544191020866243409731056, -0.00000292369433509105852898478,
-0.00000451521056375472740033756, -0.00000410111253397477487728793, -0.00000350115052254179988177425,
-0.00000460709538149641507480967, -0.00000267126967331914903975611,
-0.00000525745487178567678444082],[0.00000236689735854612036034493, 0.00005401354211893874220952461,
0.00005361209752247405858861448, 0.00000212634386834270124150520, 0.00021376343909891860873249969,
0.00021601933245713852613754946, 0.00021371351716211045026171111, -0.00021300359426054614703054113,
-0.00021061582816840062628259422, -0.00021061741798744331561309140, -0.00004934873012581498011829007,
-0.00004941124532952747715110892, 0.00020639727021737062418772768, 0.00020623915423405027010431145,
-0.00009677353186531814817350483, -0.00009669204184451884575429714, 0.00019274496593224478991485116,
0.00019244750241484230969882252, -0.00013664227172964252764696125, -0.00013648502089441260243934950,
0.00017129427512268155758649868, 0.00017086687692046981928983840, -0.00016740123771768520926327140,
-0.00016724183195677489507241509, 0.00014084176142589512905310689, 0.00014033720852142282046332022,
-0.00018914725864201307700164512, -0.00018903785169325571565573929, 0.00010124684434453062561228703,
0.00010074910169068024355220475, -0.00020305356881622632094462622, -0.00020301004742418946296539373,
0.00009964288170678496799580193, 0.00005283182990893548782156824, -0.00020276272880443801764627221,
-0.00021056322965899858831555214, 0.00000183567463261882781206812, 0.00021360306438548135651162019,
-0.00004919220042996906192781653, 0.00020586211499470417700917435, -0.00009608966261168552585539976,
0.00019172569358311930781506305, -0.00013571607969028911654801872, 0.00016983536798967206228605509,
-0.00016652900459506974457056516, 0.00013914990339860717496155151, -0.00018853503281114595440705328,
-0.00009509148455064945668849946, -0.00013447767023880887729200828, 0.00019068224041758288387835019,
0.00016834579280365134334833399, -0.00016538868811190722638709838, 0.00013744358406513892349037897,
-0.00018772671509909759704481935, 0.00009807098094741700583173993, -0.00020235617890487298382339867,
0.00005175681024165314616028444, -0.00021046484583792980389144178, 0.00000149892188486529660175635,
0.00021344673154304538102016275, -0.00004876438534439745959691431, 0.00020531907274229157784987965,
-0.00020185860419682120025196281, -0.00021034037193842330644780181, 0.00005053109160962290390042145,
0.00000114154901151524062632807, 0.00021326595474644230443495019, -0.00004822146945023048703211174,
0.00020468795438172604668480847, -0.00009387570356633688213425504, 0.00018946919872457241326539623,
-0.00013297905005305176600639838, 0.00016661659992437002041076877, -0.00016400930448360015218196850,
0.00013546794870449764699137096, -0.00018674596885885963773975271, 0.00009625906936821321464078233,
0.00018826451889321853911882232, 0.00016490430486719209789495122, -0.00013145884283336593596806205,
-0.00016261042982944199416367459, 0.00013351928059236953662096414, -0.00018574983365833599405388121,
0.00009448165801620361033304196, -0.00020135117527323337675501658, 0.00004934124374795620691447687,
-0.00021021156638463859515214560, 0.00000081579080715768989086648, 0.00021308578346141500339111508,
-0.00004765240590694399286656938, 0.00020405991050963801490027161, -0.00006369455495397468557201,
0.00004837466835853511658857920, 0.00000058155701039145530455290, 0.00021293116020920565471634400,
-0.00021010043660045379953528466, -0.00004713469428258448720671023, 0.00020352549070900421979146699,

```
-0.00009155976766124312125894047, 0.00018724422757668106320760870, -0.00013014991383698186564604748,
0.00016346134239983613071003909, -0.00016140975972011515878271060, 0.00013188131489667497116269333,
-0.00018489545185913661227289362, 0.00009300879276891316639174628, -0.00020091557636532487540574621,
-0.00012925270539438139002878791, -0.00016059350334162808765850472, 0.00016249542860370777726253255,
0.00013080809160116255314941502, -0.00018431712987497476529015483, 0.00009205034882169432243741863,
-0.00020062165658016288126604121, 0.00004776862409358900338988280, -0.00021002581909411262375704552,
0.00000047092379963368467166607, 0.00021282322220287654655768161, -0.00004674645129560488617187844,
0.00020316101713698048940141827, -0.00009080711069766315328768980, 0.00018655538713025239376010200,
-0.00021000045739016207665787739, 0.00000046849672297546872543471, -0.00004657089497451204467698416,
0.00021277618359196221364988688, 0.00020301728463624157624591027, -0.00009050865968969746243062008,
0.00018629455598901749827500801l, -0.00012891259949408129118278432, 0.00016213939723443843154981892,
-0.00016029153075424826409810242, 0.00013042108790586266005469274, -0.00018410693471924239814074831,
0.00009172144827193116976664367, -0.00020051691071988010021780180, 0.00004758042290433325182058902,
-0.00016055771981619710035878168, -0.00018430084742125576059007530, 0.00013076620099718223757916113,
0.00009204159085132512319385456, -0.00020061923390045835317342660, 0.00004779901171586560363908208,
-0.00021002901451217681092972778, 0.00000053257130397698069095870, 0.00021279526438917754135177196,
-0.00004667085025001678776123207, 0.00020311221970534766773953950, -0.00009073517813857778660600190,
0.00018649473489994885882191367, -0.00012919635532706349578119243, 0.00016243584222960663968060580,
0.00021287597035969945021005589, 0.00020342842130693659431252274, -0.00004705158616925088155906644,
-0.00009147020131272938872103728, 0.00018712152512468792787264649, -0.00013007349231749501181076845,
0.00016333493692558394240130215, -0.00016135866841565510974333686, 0.00013178471282205921587260544,
-0.00018487241818889102863525398, 0.00009295264355714057569624736, -0.00020091399527599117192863676,
0.00004837685879890161398360696, -0.00021010732045492251196354416, 0.00000063455943585538114612654,
0.00013332910436741305130678437, 0.00009432959386507843509324378, -0.00018573553760926076188104983,
-0.00020135596579710785523388561, 0.00004924223542495276707137153, -0.00021022297660932989880194777,
0.00000077161057427401206453959, 0.00021300482875508694544791632, -0.00004765008377741901580580339,
0.00020391582872973987659288423, -0.00009260344197997834000176537, 0.00018807822340537842255824008,
-0.00013141456634410987320930875, 0.00016470163029530234064662375, -0.00016257544129588528568006178,
-0.00004835824054278155525185903, -0.00009395271273304020055467528, 0.00020449916076300010204028212,
0.00018921915429833945109829030, -0.00013301085392361386905077381, 0.00016633149418526828604465717,
-0.00016402030784208764175025961, 0.00013517463094014407439837078, -0.00018675667971291304170410473,
0.00009598120127437145224658793, -0.00020187596179295096057021430, 0.00005028706674856419690539805,
-0.00021035723530740766534785635, 0.00000094583154636148802877770, 0.00021316146855272514386119009,
-0.00018777507555534154841392347, -0.00020239131481470396260909095, 0.00009766176270854985360693468,
0.00005136112093351594625563125, -0.00021048788038304630976411957, 0.00000114226841557361437610669,
0.00021332173759518082822167484, -0.00004905535865031053770375646, 0.00020508918421738761539950546,
-0.00009530040292007477210813743, 0.00019037055561044286216935972, -0.00013460931789883412673484997,
0.00016797744304176165136728571, -0.00016546581892893483961039236, 0.00013704335249608529481192554,
0.00020559627324261584512508916, 0.00019135688056071982867181736, -0.00009642934573468223786660292,
-0.00013595332387582138920605057, 0.00016938709923676527227369426, -0.00016668061398428593578024959,
0.00013864631827251148207640064, -0.00018862763318626922645279698, 0.00009910871380518841938615793,
-0.00020281861622180536244841598, 0.00005229500786014602598016435, -0.00021059267657396406846215253,
0.00000133035900151194188105745, 0.00021346138940223207953843232, -0.00004962866799930752735939277,
0.00010009566851140094996281543, 0.00005294204580640715196004215, -0.00020308674876352617076911652,
-0.00021065285855841956439739315, 0.00000148279723107899029957380, 0.00021355977879903943815291523,
-0.00004998186877311153739674582, 0.00020594420874012721246922031, -0.00009715292863899149041363029,
0.00019202847770110375636744282, -0.00013682202465784693012906814, 0.00017034449686958069757405920,
-0.00016746577030639684486798546, 0.00013973528562291015974426711, -0.00018917456231951502304555746,
-0.00009734081993704317458445535, -0.00013706416560277417037924663, 0.00019228550987648464058211994,
0.00017070634048055346163909329, -0.00016768627437649998848309685, 0.00014014736108407480227010244,
-0.00018932159698878040345396134, 0.00010047729859783213627114073, -0.00020314815192578876304300983,
0.00005321273963208498727255405, -0.00021065611869494136449386390, 0.00000159047386754816555851131,
0.00021360300032390687062673618, -0.00005004284533285944305000884, 0.00020608198701018627684890261,
-0.00020298651286356650001382884, -0.00021059862546931836565361484, 0.00005308726480185237649178642,
0.00000166196930536233599082240, 0.00021358597912658565907972896, -0.00004977661117589493350484020,
0.00020599168749371279713185345, -0.00009694075019949954417301102, 0.00019209369333580609108882986,
-0.00013662301018302843466471763, 0.00017042490890101463871936227, -0.00016729312921199318195918104,
0.00013983095900683837264563691, -0.00018903533268685017735763476, 0.00010021168589373279458366872,
-0.00013554822352933954524406577, -0.00016633245667899010790218017, 0.00016955430576301742722974963,
0.00013885091170556801551047577, -0.00018834909896046274444675328, 0.00009936136322858228901895583,
-0.00020261962199421507742520820, 0.00005261064116727419371841168, -0.00021048574935315036049839066,
0.00000171337390899023825080059, 0.00021351319371197845144866312, -0.00004919850242101386951368638,
0.00020569111695894478708881992, -0.00009599204590157232289300560, 0.00019148924600234513136200187,
0.00021339791560930228796200137, -0.00021033137516208007472898189, 0.00000175966114732989418704284,
-0.00004837721882274796121993846, 0.00020523084996400489817512669, -0.00009462322700216664877071598,
0.00019057260081400051107691773, -0.00013399004480561230081420154, 0.00016824030929786629217151384,
-0.00016493916311344046640371963, 0.00013737511321888575574305669, -0.00018735823757851658355119695,
0.00009807884414871537084174014, -0.00020209691050995912033755308, 0.00005187996759700072140495189,
0.00016669626682979941695479653, 0.00013564600188451338743517161, -0.00016331554584148575727520214,
-0.00018620541242146896178215332, 0.00009658019822995888290558669, -0.00020149203778585097405866289,
0.00005102759603463510423545027, -0.00021015590352159302920710682, 0.00000181251451853404934958251,
0.00021326009234269244023950163, -0.00004742244181526253778233976, 0.00020468619001289613560864988,
-0.00009302998797134073090571438, 0.00018949218651015582465940712, -0.00013217460481848797510320837,
0.00000188183096733821720918046, -0.00004646306178227798219142544, 0.00021312320187091661548121901,
0.00020414536535904166366164891, -0.00009144059856885732679929180, 0.00018842107421555658302442238,
-0.00013036638263550800022685938, 0.00016516932778448990284080045, -0.00016169910417474810376131655,
0.00013394203931830443027317656, -0.00018505853569537650173820742, 0.00009511098788545195883680525640,
-0.00020089172519817714238771211, 0.00005020161802245859730946947, -0.00020998324462182294566740814,
-0.00016032597487864515706929014, -0.00018408506926736689060693231, 0.00013253533509886359221911778,
0.00009390970411895653872577650, -0.00020038261517049631003396792, 0.00004954477677215978978237515,
-0.00020983726687291479873394386, 0.00000197578821772630547676697, 0.00021301057107211401734671308,
-0.00004562627780974468425087207, 0.00020369581070737321592915847, -0.00009008022043937425173479316,
0.00018753014173850509000232578, -0.00012882727487069587095531997, 0.00016390236616676046358352659,
0.00021294172529952472413747322, 0.00020341065109891282784541910, -0.00004502166004309501721363310,
-0.00008913988972122294037920037, 0.00018696181859359240753568066, -0.00012777871683883405525342103,
0.00016309694613150704160624282, -0.00015939577032451232260247542, 0.00013165059781623895062384333,
-0.00018342677777614482836389398, 0.00009317197016279605257822044, -0.00020003820446215492788455403,
```

```
0.00004917235129883729257901925, -0.00020973824458595628230338226, 0.00000209747380690658305900007,
0.00013143059119717278809049427, 0.00009301959721049492247051810, -0.00018317858475078900928889603,
-0.00019990783628202538199300498, 0.00004915206393052292354033800, -0.00020969984424572089831695365,
0.00000224057723819562692061454, 0.00021292932978375189549680169, -0.00004472865265418797314431187,
0.00020333744884580098898942124, -0.00008875081727962814482409820, 0.00018680831799521921822264392,
-0.00012737006056659394451040235, 0.00016288256973613317985481730, -0.00015904222592714236392610894,
-0.00004478461487772766256567833, -0.00008896441909145279562567271, 0.00020349091060582237284332097,
0.00018709768610017716130092258, -0.00012765611819718898720804612, 0.00016329711683072274100089749,
-0.00015931336728019438722311896, 0.00013191433264510263648726374, -0.00018337456214367854005070635,
0.00009348155509948895258969664, -0.00020000939188601177557910726, 0.00004949306625213800757759341,
-0.00020972710623493162466514372, 0.00000238983050735444537608729, 0.00021297718916038889478553608,
-0.00018398315763743915704844789, -0.00020032675406499094263902683, 0.00009449216034864956273442438,
0.00005014870894073889949667244, -0.00020981572132553852100440850, 0.00000252865124875403699013774,
0.00021307960767526475998924484, -0.00004517421321322968294281405, 0.00020385063315851497875441645,
-0.00008974154212259016652562110, 0.00018778976759760101469982907, -0.00012858803279102554682766790,
0.00016428179261177328789912677, -0.00016016452372189310328527700, 0.00013303259613173861407829646,
-0.00009095873254367217101545168, -0.00013002181463364067626545251, 0.00018878250468945596720997393,
0.00016569080202047639280814317, -0.00016146599517999967333264921, 0.00013462003300271314703770442,
-0.00018491239573189665462704667, 0.00009590359714470447229615890, -0.00020081238310014058756104716,
0.00005102676789396104635714718, -0.00020995270643397084863249202, 0.00000264452492808631948819384,
0.00021322220444996276391205758, -0.00004583031933930512133099078, 0.00020436418097369337125608635,
-0.00020139457054747816238185476, -0.00021011837168406554765889838, 0.00005199824380140341512730393,
0.00000272280146821866843579368, 0.00021338405786834859704896583, -0.00004665295441673965732816767,
0.00020495498505802614542213591, -0.00009243352129873707472614702, 0.00018992737085119523830073485,
-0.00013174350693034869328942871, 0.00016731337563349006755317183, -0.00016302382379238214318677158,
0.00013643967874164272336204307, -0.00018602406973527294193650194, 0.00009750611236171419617460576,
0.00019105156639495987505941721, 0.00016890490087334553408887627, -0.00013350205524091196627317624,
-0.00016460890798362668775191298, 0.00013821707486982872888948981, -0.00018715448846857901167052562,
0.00009905634124965563037934491, -0.00020198824848110914710781760, 0.00005291133229075066968152136,
-0.00021028927693523695495805836, 0.00000274215455054867476503079, 0.00021354085513318514727779118,
-0.00004753000996580814083592587, 0.00020553385861102628408975113, -0.00009395478661697239395125814,
0.00005361925186353092243020033, 0.00000268632338141147337766401, 0.00021366857053165268470577842,
-0.00021044173118879013839946368, -0.00004834422379271559745261139, 0.00020601242450934444377236288,
-0.00009530652655494274011482858, 0.00019198380347336301888529209, -0.00013504261160062464952429095,
0.00017022320317893461409999856, -0.00016598979521232747752490511, 0.00013968135484984135763486535,
-0.00018813872729791416139703830, 0.00010031552524288051128284471, -0.00020250774239127889504892555,
-0.00013614143289071532218724447, -0.00016696641590870862350767800, 0.00017106461539639585727259896,
0.00014060608245580221136589172, -0.00018883521983308152467420649, 0.00010108788072778741536951463,
-0.00020287959883068579680816257, 0.00005400895300013944418104916, -0.00021055530532653992238471818,
0.00000255675348328319715756067, 0.00021374712003927040580679442, -0.00004898045466993483502190360,
0.00020631647121319583097591444, -0.00009629314020292675021378548,
0.00019257997464974666631393196],[0.00024221422419563195757054308, 0.00023512113755097093967770527,
0.00023058057863197106425283411, 0.00023748759163156555468399134, 0.00003642079935915664000105013,
0.00000004876154627997606448055, 0.00003573772057353782275867377, -0.00000002088286326493736915215,
0.00003643061773194065450709109, 0.00003569517911954019377326897, 0.00023527044183071184400672660,
0.00023063332513228700505585056, 0.00007358337887977286866352661, 0.00007220446304952278591526155,
0.00021542909687234535072962449, 0.00021114630243880184068579198, 0.00011169629608104612384159027,
0.00010960034796322819868226450, 0.00018572736855210116978937007, 0.00018200969597622135482707650,
0.00014977057135253255752520285, 0.00014694921814450832191559226, 0.00015003603417210176467627702,
0.00014701794287069622430355609, 0.00018543487975514105141165710, 0.00018191940662472020034771958,
0.00011189019276786302020849917, 0.00010963275076850091259601794, 0.00021517429985181928792613293,
0.00021106024557111693722155632, 0.00007368546393960541179488088, 0.00007219722349858664133415048,
0.00019882741133448443050790733, 0.00021717139964537093962601566, 0.00006793431900729370216413666,
0.00003358734532889100704361116, 0.00022362899337138533338228963, 0.00003368256367660931566675175,
0.00021713104697561533816020651, 0.00006805053168791414203251527, 0.00019874982164681107205984889,
0.00010329023768253383329385198, 0.00017129906074155973219434734, 0.00013847623746089191281057051,
0.00013835141970931563249261331, 0.00017140640187023446923929304, 0.00010316280799831033482098774,
0.00017871586924658074677953190, 0.00015400894491171836209707224, 0.00009300516107585952724906564,
0.00012467183885985752599470655, 0.00012437208778106738924287833, 0.00015429306820766521858578901,
0.00009273112439351300425195018, 0.00017893891052621962111345011, 0.00006106174678821117555223980,
0.00019540257626958067605632552, 0.00003018844253601195144168408, 0.00020116622036635067394608201,
0.00003033399547645973360981107, 0.00019527973369703438663753936, 0.00006127981404725383425199925,
0.00005184462628418332080142486, 0.00002562935004478196328249646, 0.00016610642401736482411979401,
0.00017096096314956807891982660, 0.00002582048400143181171952945, 0.00016591982488141181466961838,
0.00005215152293647389768410996, 0.00015181650224050684456524041, 0.00007913788028676893213358284,
0.00013080666634865718172577032, 0.00010606201718466357910262976, 0.00010561992236396534970265665,
0.00013123095997358502845867023, 0.00007874056746028607154264806, 0.00015215293565708012404023852,
0.00006222040683111706386304585, 0.00008335959549927594771506767, 0.00010258632797025758385221772,
0.00008281781030143020561583300, 0.00010310196836561219915252841, 0.00006173032279876991220736449,
0.00011949298694665030104531539, 0.00004063789668697304286251737, 0.00013040326174395598201986857,
0.00002008545666738634850921955, 0.00013417078693930139946313695, 0.00002031544022311951151228693,
0.00013017928887589910106996338, 0.00004101619549416230968083563, 0.00011908714891399693085790451,
0.00008966129511665799339120708, 0.00009220505754501783900883849, 0.00001403056240262761242129474,
0.00001376989274697555899675417, 0.00008942866614871386606554216, 0.00002830234007726331423157248,
0.00008178486589571475084449459, 0.00004290405945425181214207197, 0.00007043310183359100635439726,
0.00005743847000855366427595275, 0.00005684299892959915643598015, 0.00007098765321529926426193102,
0.00004235476262650960586854040, 0.00008221259424061854988261683, 0.00002787253843428671366435917,
0.00003558242376847433067945520, 0.00002869342762609467504695757, 0.00002930084053037463522202975,
0.00003613072621715948874081617, 0.00002135823672786075472077895, 0.00004175360325406188895941417,
0.00001403897976335879906484988, 0.00004545334004177860255831167, 0.00000692532427039974761851914,
0.00004668015966986112082279942, 0.00000720761288611706515577338, 0.00000452350843280735238899716,
0.00001449990051279543850531236, 0.00004134314658568343837669740, 0.00002193448472471466867070311,
-0.00000018536275476188765807097, -0.00000063555088312147990051593, -0.00000069105112124982154233108,
0.00000010937065147812057006363, 0.00000014112585398400017200403, -0.00000067819484557037945251883,
0.00000012223552315333760713112, -0.00000062520755138185074585962, 0.00000003739673396732308540177,
-0.00000054985479003812424010330, -0.00000011406121089387746799567, -0.00000045335505234210048940157,
-0.00000030872975031862248642604, -0.00000033173271494498099654726, -0.00000049961619875163898049847,
```

```
-0.000029763912143342309167201 59, -0.000022243324884897918640013 63, -0.000036338247952775007984602 82,
-0.000042335382678952864269304 73, -0.000014688226203887656337999 76, -0.000046404162964593085086671 69,
-0.000007289136268972304960492 12, -0.000047896923463989600152929 81, -0.000006991119792288161114866 48,
-0.000046564606718138508217527 67, -0.000014220273576558650313302 40, -0.000042653861853473191154476 06,
-0.000021689717412228976824682 68, -0.000036795525053136489662241 47, -0.000029218171337606940435696 120,
-0.000013820538721458081586092 31, -0.000028030804231963211969880 23, -0.000090609907614930409276363 11,
-0.000082964242214905468205017 60, -0.000042659617774180328341696 73, -0.000071537583465691332793023 88,
-0.000057335334403057225676099 42, -0.000057827807958285095161436 03, -0.000071140587145839461954976 99,
-0.000043176210719032136100669 49, -0.000082698516061855710784084 25, -0.000028479841217127037 66895 4751,
-0.000090480239178568119665958 81, -0.000014113277127250951908802 10, -0.000093271216602289336024093 90,
-0.000103183310963096165624452 99, -0.000119849260439792003481683 51, -0.000062449764207044608076840 89,
-0.000041177624989928871749970 96, -0.000131038936226919328770224 03, -0.000020395811942936689631616 23,
-0.000135021061231399325426594 25, -0.000020116148216395169642323 54, -0.000131140603035928560160527 92,
-0.000040758726870140146033151 59, -0.000012006568355820456989353 73, -0.000061979878778764693870724 ,
-0.000103520745533987231770320 37, -0.000083231926311800040777996 72, -0.000083666474728815131952155 97,
-0.000166619823953045317161889 01, -0.000152547202844148191034842 62, -0.000051914591488609472980531 73,
-0.000078908039779458573770992 70, -0.000131524543056303074668386 69, -0.000105914501930052509705987 30,
-0.000106291312458396249658772 18, -0.000131240845106607629698067 25, -0.000079325375091007817979736 62,
-0.000152372194950520337434693 44, -0.000052294499076472725811911 80, -0.000166540757399596966613006 25,
-0.000025895538119338655175767 35, -0.000171564781270453305984935 44, -0.000025635868376721571381615 03,
-0.000093155883081574999233717 69, -0.000061403788282232462081940 89, -0.000179029775213529311356924 77,
-0.000195638862070728102374173 04, -0.000030401257742078110005157 22, -0.000201517953725380098966429 85,
-0.000030167589487220448265962 06, -0.000019570217940511938615154 308, -0.000061069696650797516781641 ,
-0.000179173364936161733169957 633, -0.000092794816359223432269270 89, -0.000154479979821783478628402 51,
-0.000124514742385630828355042 03, -0.000124836124455014079500519 97, -0.000154242059745662049401182 53,
-0.000067873794149578826309827 94, -0.000103108468653628453765534 74, -0.000198923174889403597343298 66,
-0.000171506104005963977941762 18, -0.000138321328339371837605846 , -0.000138588785434950133568976 4,
-0.000171307917119285156278343 80, -0.000103410010700761255137553 05, -0.000198803166502448900318858 47,
-0.000068155575614372424279006 90, -0.000217220258184994580467550 02, -0.000033739878258765954954865 46,
-0.000223733747428366317420989 07, -0.000033537321414154678258343 73, -0.000217273313897939764028841 01,
-0.000210931449030280355723745 99, -0.000230450091022772968543667 3, -0.000072290136742081415913801 800,
-0.000035783058717487360640150 280, -0.000237349557786539999649708 11, -0.000035615832168702386199658 77,
-0.000230494924131106903881366 37, -0.000072065670100675586432631 41, -0.000211030471896522743482232 93,
-0.000109454825798166230214382 43, -0.000181944325591348795298843 32, -0.000146806505509651475868682 09,
-0.000147018695666590132997439 822, -0.000181784605491614684875572 8, -0.000109692884853249765918526 4,
-0.000215020532053137940989989 70, -0.000185387538110508807986942 51, -0.000111592099608838739993599 28,
-0.000149646404401366692714370 20, -0.000149798793880479507410752 32, -0.000185269871308998263208928 09,
-0.000111761628336928158563429 17, -0.000214944873331641816319705 515, -0.000073648030638420928606184 55,
-0.000234811494972679080453503 71, -0.000036452146208368745050089 34, -0.000241830433096581229150678 90,
-0.000036323589394508721303177 91, -0.000234846973282818957779188 969, -0.000073485700701658471603074 10,
-0.000072176374590906451655415 72, -0.000035721231320831371713626 22, -0.000230131938938934777231712 37,
-0.000236997503213458174733951 90, -0.000035633750485663586717813 98, -0.000230155414636716436418259 88,
-0.000072080514150000353544925 36, -0.000210734412886083062469225 19, -0.000109440150289692402808816 58,
-0.000181699478742842823264860 84, -0.000146733884286734161993617 95, -0.000146819743673437329944207 0,
-0.000181630395914419095169944 91, -0.000109535301582904212483286 78, -0.000210687147583194374052609 21,
-0.000171021437240164635249645 15, -0.000138194850271111370704560 37, -0.000138183243913021915972097 48,
-0.000171008629498478164883759 62, -0.000103098248442304172211789 19, -0.000198324595092114739670885 46,
-0.000067930961956484461612519 60, -0.000216593948152299245815677 33, -0.000033618172173470924079825 15,
-0.000223038888427685695701324 64, -0.000033573167025571608312883 85, -0.000216602384699815823584923 00,
-0.000067905214515320174426850 35, -0.000198337574947620310771706 50, -0.000103083555091668143076882 869,
-0.000030221292761372498111502 32, -0.000030223541915837479347912 99, -0.000200502242361409992871257 30,
-0.000194717992367168162091278 78, -0.000061121194592830279551133 60, -0.000178312102294491423102457 13,
-0.000092768431791261210571722 38, -0.000153765806330807830448748 06, -0.000124326187134063468990585 74,
-0.000124255309657460028677222 75, -0.000153817929828712338709978 40, -0.000092696862044124310638093 52,
-0.000178340297394620841704010 46, -0.000061074195564891515792919 91, -0.000194728337379650986906781 67,
-0.000105698725169096595659541 66, -0.000130725427215647345413618 30, -0.000105536379972257421313436 29,
-0.000078729817698814174935749 43, -0.000151512566693350064642512 03, -0.000051868861945717958039040 60,
-0.000165388682054949637907290 13, -0.000025667572213476848439992 885, -0.000170267157940042542646486 01,
-0.000025707115088511623568088 77, -0.000165353689450224830602767 52, -0.000051989833988271135986614 09,
-0.000151433348558639686116595 753, -0.000078892837849053116190378 22, -0.000130597503462094805329329 89,
-0.000133503400662298544283521 08, -0.000129642710487491014115235 54, -0.000020204157699034933765285 67,
-0.000040862366079412568327378 09, -0.000118735693735217499845090 16, -0.000061991158237092425209967 26,
-0.000102406125739720781564132 12, -0.000083019601747351558619983 37, -0.000082755957004652542781213 13,
-0.000102624054330813284461459 97, -0.000061732668493234520699389 94, -0.000118880409063791036359645 25,
-0.000040668082077055334535870 68, -0.000129711611617909196780482 06, -0.000020125175055288618711311 22,
-0.000056787096049194121338650 80, -0.000042357364892694850436703 77, -0.000070596111005921116047373 55,
-0.000081700854554484839934076 77, -0.000027901833702166238581135 47, -0.000089070375411587305509136 39,
-0.000013809241698366188567286 34, -0.000091623945557558656728990 02, -0.000013923808768326493968243 12,
-0.000088955251780050846772962 55, -0.000028166261592225878715401 93, -0.000081472179118300197477216 26,
-0.000042713091837224629048093 42, -0.000070271623507795190843004 67, -0.000057161505685990522023301 46,
-0.000007107201868523050370123 24, -0.000014388738756376780121614 20, -0.000044848806800139637664462 39,
-0.000041069039051024921864203 18, -0.000021798349217340775191590 83, -0.000035424693400998022105048 01,
-0.000029116643297044404720521 14, -0.000028625489752630229651651 43, -0.000035870464574476619828286 02,
-0.000021347655091862731446036 16, -0.000041399931322173278943395 44, -0.000014060545870533112423894 64,
-0.000045022854451578803603589 71, -0.000006962491896330101524500 73, -0.000046232910931898040242023 25,
0.000000224679115270076657042 46, 0.000000481674926062977296473 19, 0.000000488995186273133720064 63,
0.000000323634629591537532302 31, 0.000000747066687273364509514 79, 0.000000151861058105609518307 98,
0.000000933699330495662247969 36, -0.000000015975259314781658701 29, 0.000000988500669697860421977 72,
-0.000000058092519680199423859 24, 0.000000925770029497265418898 61, -0.000000048200787602423613798 04,
0.000000797976233256873854820 19, 0.000000041452432646459117052 69, 0.000000647374912147137865058 30,
0.000046800357440172450623572 30, 0.000042899674371040838569560 40, 0.000014276460836239682127136 ,
0.000021704914700296568981366 73, 0.000037003336859304344268282 31, 0.000029198343676355583656712 58,
0.000029904953308716394639460 43, 0.000036311309087926970393198 77, 0.000022312166479834413498114 37,
0.000042345859536182621485605 33, 0.000014697270056984819468432 27, 0.000046492102782998422387181 20,
0.000007260225996523574819740 9, 0.000048072559897323919873052 7, 0.000007077741937156103823923 8,
0.000043281010739282734932527 33, 0.000028506991064106836862015 67, 0.000082595090568763512330063 84,
0.000090465043098142102276360 00, 0.000014089177337854421600822 46, 0.000093379343382774256479041 12,
```

```
      0.00001390172785790030196692232, 0.00009082805648641118900082914, 0.00002806563960136081002519695,
      0.00008323702796946372977921619, 0.00004262872853756081065234110, 0.00007179579171989889567560145,
      0.00005724008582118376498634044, 0.00005801929525829632245314746, 0.00007101209043807231804674301,
      0.00012038333834286814294761009, 0.00010383306764068984972583520, 0.00006192240669891687482838394,
      0.00008309469525665875077810596, 0.00008390578792607722500955514, 0.00010300121812758039553832579,
      0.00006258715875818359006960145, 0.00011969106390390556713971409, 0.00004122096533532975401806220,
      0.00013098303125910060315181815, 0.00002037599858503987373427313, 0.00013511635305782260002734063,
      0.00002019407080782172699169597, 0.00013137857324053720670109513, 0.00004078093529010286559435133,
      0.00005234954471178215960457808, 0.00002587883134899454044136613, 0.00016649349418400413627011636,
      0.00017168185862034981243803844, 0.00002571317762172624241329491, 0.00016689317445149211391504374,
      0.00005193466288569443666412601, 0.00015290857304592109277077694, 0.00007884675684891602855716014,
      0.00013188042058524072496113189, 0.00010577246912910297216552785, 0.00010656630312335742445401660,
      0.00013105468082139063346232277, 0.00007948646572211354763108637, 0.00015221385474413457556883134,
      0.00009275223431934548432714616, 0.00012440355572374795184269203, 0.00015485901300309637775888538,
      0.00012512830010899064921668644, 0.00015409710537769833295709820, 0.00009332802808153728359348567,
      0.00017891747244875335883310119, 0.00006146429238704890509435719, 0.00019563624679433259164612613,
      0.00003038600629161280648467096, 0.00020167483423115019087329403, 0.00003024702323752094363881228,
      0.00019600983424611289154294858, 0.00006109853669347578390691478, 0.00017956355425588266033418150,
      0.00021729386764661110095428476, 0.00022394623173360359274369769, 0.00003362128005204847520015141,
      0.00003372421628176038894833880, 0.00021761287056678890782304703, 0.00006792007336981402225700027,
      0.00019932597152122770358534387, 0.00010310376552645847273229074, 0.00017188644896625095018563578,
      0.00013827097185156560221752642, 0.00013887814927907385104816074, 0.00017124231788059795437810517,
      0.00010357934222891635389022374, 0.00019877640983226803924992232, 0.00006821455257841429445538062,
      0.00018230925383158519165367750, 0.00014728808845052273408977872, 0.00014683821948869725635029859,
      0.00018182656923916606147434427, 0.00010984671095598225567298994, 0.00021102242965774522750511955,
      0.00007234097598022849436374959, 0.00023063008265034889185222944, 0.00003576519103914519034058406,
      0.00023763783581554187422126001, 0.00003570603085499142020356120, 0.00023087139189595877854579986,
      0.00007213616985492927720455963, 0.00021143688368655628919551281, 0.00010950145260021515328328207]]
00061
00062 def main():
00063     test_finalSF()
00064     test_bFieldGeneratedByOptSF()
00065
00066 if __name__ == "__main__":
00067     main()
```

8.22 subfunctions/Tester.py File Reference

Classes

- class Tester.Tester

Namespaces

- namespace Tester

8.23 Tester.py

Go to the documentation of this file.

```
00001 class Tester():
00002     def __init__(self):
00003         self.gaußLegendre = []
00004         self.reducedSF = []
00005         self.calcWeightsGauss = []
00006         self.matElementsShouldGetValue = []
```

Index