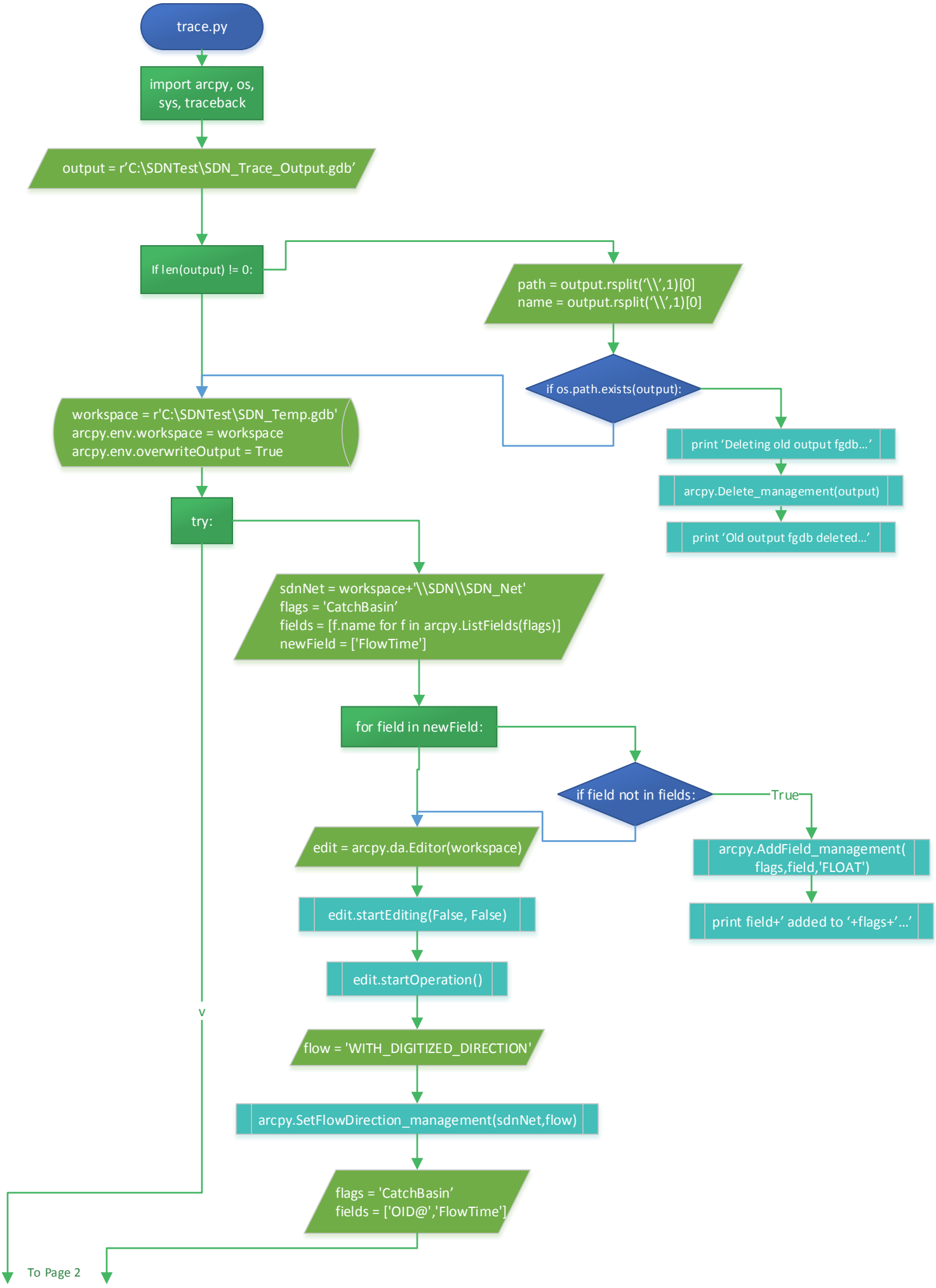
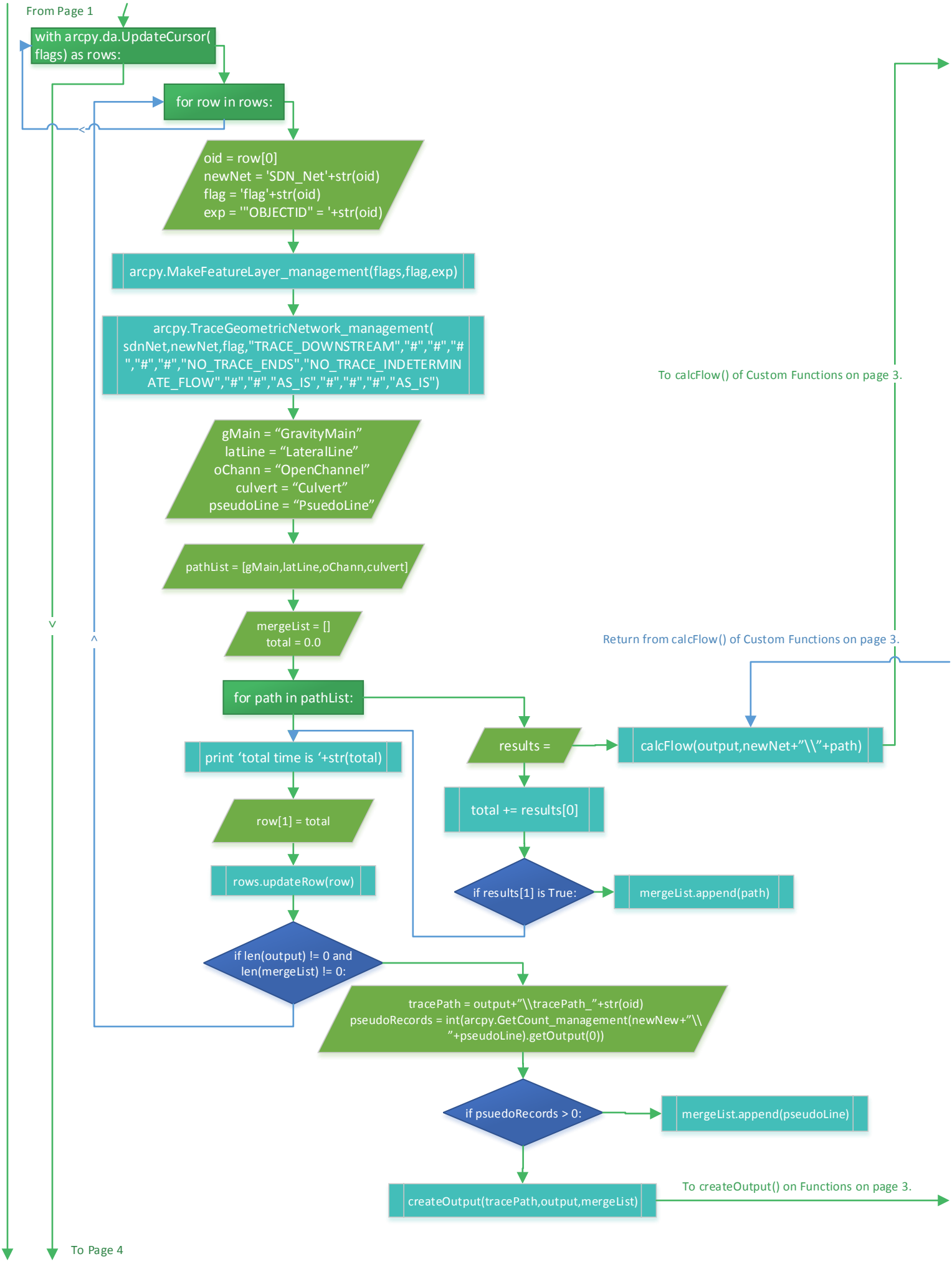
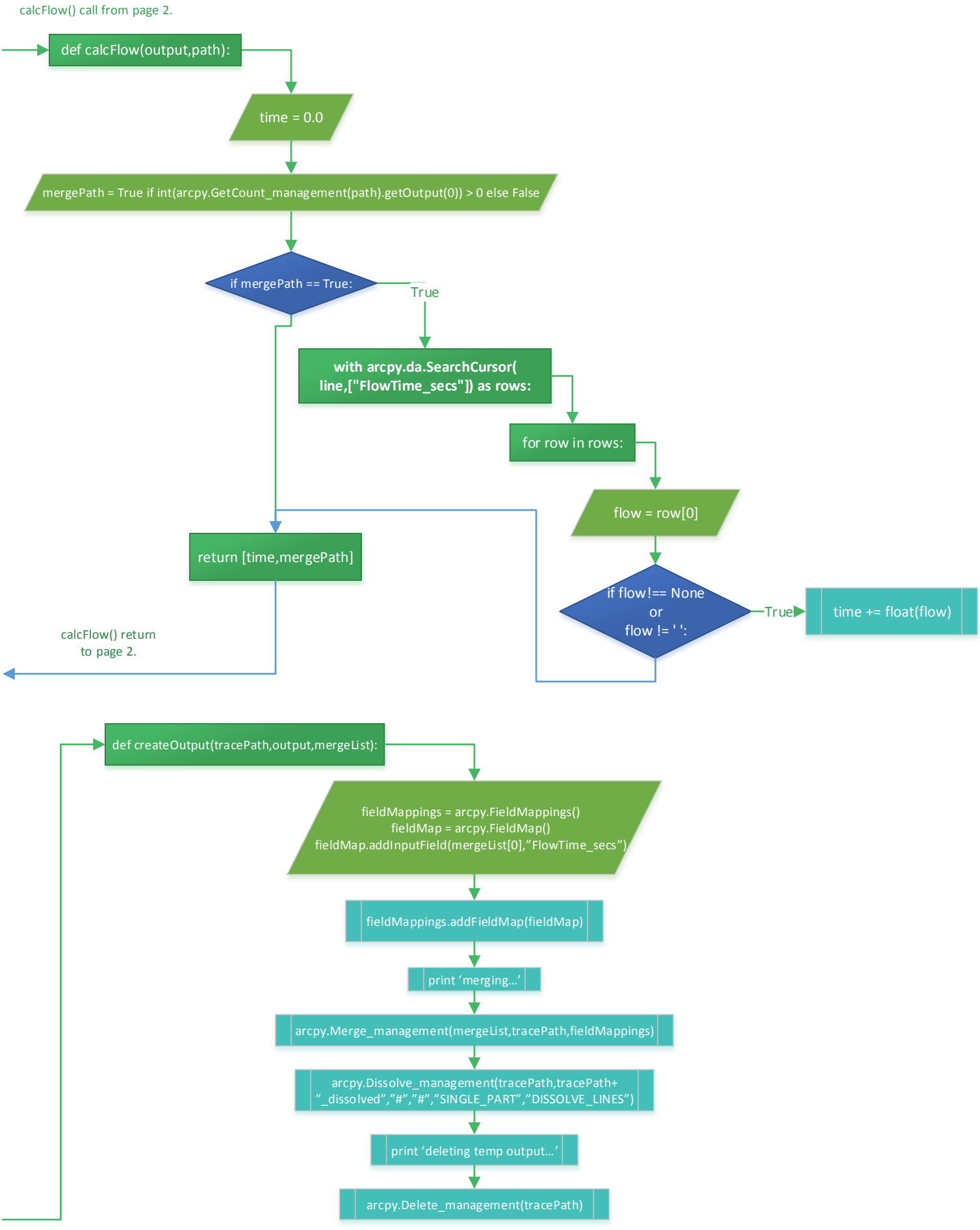
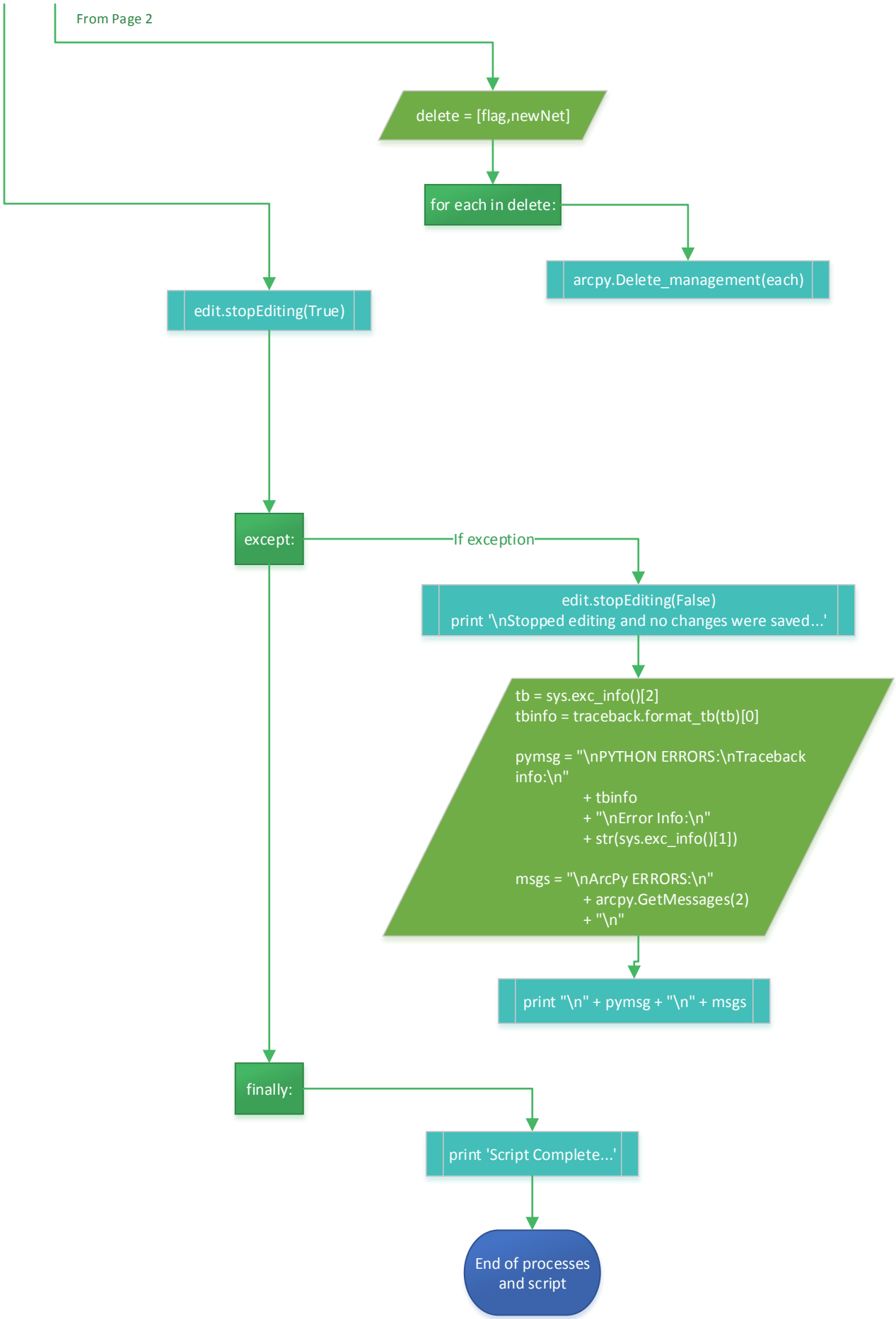

trace.py
import arcpy, os, sys, traceback
output = r'C:\SDNTest\SDN_Trace_Output.gdb'
If len(output) != 0:
path = output.rsplit('\\',1)[0]
name = output.rsplit('\\',1)[0]
if os.path.exists(output):
print 'Deleting old output fgdb...'
arcpy.Delete_management(output)
print 'Old output fgdb deleted...'
workspace = r'C:\SDNTest\SDN_Temp.gdb'
arcpy.env.workspace = workspace
arcpy.env.overwriteOutput = True
try:
sdnNet = workspace+'\\SDN\\SDN_Net'
flags = 'CatchBasin'
fields = [f.name for f in arcpy.ListFields(flags)]
newField = ['FlowTime']
for field in newField:
if field not in fields:
True
arcpy.AddField_management(flags,field,'FLOAT')
print field+' added to '+flags+'...'
edit = arcpy.da.Editor(workspace)
edit.startEditing(False, False)
edit.startOperation()
flow = 'WITH_DIGITIZED_DIRECTION'
arcpy.SetFlowDirection_management(sdnNet,flow)
flags = 'CatchBasin'
fields = ['OID@','FlowTime']
v
To Page 2


From Page 1
with arcpy.da.UpdateCursor( flags) as rows:
for row in rows:
oid = row[0]
newNet = 'SDN_Net'+str(oid)
flag = 'flag'+str(oid)
exp = '"OBJECTID" = '+str(oid)
arcpy.MakeFeatureLayer_management(flags,flag,exp)
arcpy.TraceGeometricNetwork_management( sdnNet,newNet,flag,"TRACE_DOWNSTREAM","#","#","#","#","NO_TRACE_ENDS","NO_TRACE_INDETERMINATE_FLOW","#","#","AS_IS","#","#","#","AS_IS")
gMain = "GravityMain"
latLine = "LateralLine"
oChann = "OpenChannel"
culvert = "Culvert"
pseudoLine = "PsuedoLine"
pathList = [gMain,latLine,oChann,culvert]
mergeList = []
total = 0.0
for path in pathList:
print 'total time is '+str(total)
row[1] = total
rows.updateRow(row)
if len(output) != 0 and len(mergeList) != 0:
tracePath = output+"\\tracePath_"+str(oid)
pseudoRecords = int(arcpy.GetCount_management(newNew+"\\"+pseudoLine).getOutput(0))
if psuedoRecords > 0:
mergeList.append(pseudoLine)
createOutput(tracePath,output,mergeList)
To createOutput() on Functions on page 3.
results =
calcFlow(output,newNet+"\\"+path)
total += results[0]
if results[1] is True:
mergeList.append(path)
To calcFlow() of Custom Functions on page 3.
Return from calcFlow() of Custom Functions on page 3.
To Page 4


calcFlow() call from page 2.
def calcFlow(output,path):
time = 0.0
mergePath = True if int(arcpy.GetCount_management(path).getOutput(0)) > 0 else False
if mergePath == True:
True
with arcpy.da.SearchCursor( line,["FlowTime_secs"]) as rows:
for row in rows:
flow = row[0]
if flow!== None or flow != ' ':
True
time += float(flow)
return [time,mergePath]
calcFlow() return to page 2.
def createOutput(tracePath,output,mergeList):
fieldMappings = arcpy.FieldMappings()
fieldMap = arcpy.FieldMap()
fieldMap.addInputField(mergeList[0],"FlowTime_secs")
fieldMappings.addFieldMap(fieldMap)
print 'merging...'
arcpy.Merge_management(mergeList,tracePath,fieldMappings)
arcpy.Dissolve_management(tracePath,tracePath+ "_dissolved","#","#","SINGLE_PART","DISSOLVE_LINES")
print 'deleting temp output...'
arcpy.Delete_management(tracePath)


From Page 2
delete = [flag,newNet]
for each in delete:
arcpy.Delete_management(each)
edit.stopEditing(True)
except:
If exception
edit.stopEditing(False)
print '\nStopped editing and no changes were saved...'
tb = sys.exc_info()[2]
tbinfo = traceback.format_tb(tb)[0]
pymsg = "\nPYTHON ERRORS:\nTraceback info:\n"
+ tbinfo
+ "\nError Info:\n"
+ str(sys.exc_info()[1])
msgs = "\nArcPy ERRORS:\n"
+ arcpy.GetMessages(2)
+ "\n"
print "\n" + pymsg + "\n" + msgs
finally:
print 'Script Complete...'
End of processes and script