******* Cover Page *******

Class: CV

Name: Frank Yournet Project: Project 5

Project Name: Image Compression via Distance Transform

Language: C+-

Due Date: 10/20/2024 before 12:00AM Submit Date: 10/20/2024 before 9:00PM

Top Level algorithm steps

VI. main (...)

step 0: inFile, prettyPrintFile, skeletonFile, deCompressedFile, logFile ← open via argv []

numRows, numCols, minVal, maxVal ← read from inFile

dynamically allocate ZFAry and skeletonAry with extra 2 rows and 2 cols

distanceChoice ← get from argv [2]

Step 1: setZero (ZFAry)

setZero (skeletonAry)

Step 2: loadImage (inFile, ZFAry)

prettyPrint (ZFAry, prettyPrintFile) // with caption "** Below is input image**"

Step 3: distanceTransform (ZFAry, distanceChoice, prettyPrintFile, logFile)

Step 4: compression (ZFAry, distanceChoice, skeletonAry, skeletonFile, prettyPrintFile, logFile)

Step 5: close skeletonFile

Step 6: reopen skeletonFile

Step 7: setZero (ZFAry)

Step 8: loadSkeleton (skeletonFile, ZFAry, logFile)

prettyPrint (ZFAry, prettyPrintFile) // with caption "** Below is the loaded skeletonwith choice =**"

Step 9: deCompression (ZFAry, distanceChoice, prettyPrintFile, logFile) // Perform decompression

Step 10: deCompressedFile ← output numRows, numCols, minVal, maxVal

Step 11: binThreshold (ZFAry, deCompressedFile)

Step 12: close all files

Source Code

#include <iostream>

#include <fstream>

#include <string>

#include <sstream>

using namespace std;

```
class DistanceSkeleton{
  public:
     int numRows;
     int numCols;
     int minVal;
     int maxVal;
     int newMinVal;
     int newMaxVal;
     int** ZFAry;
     int** skeletonAry;
     int distanceChoice;
     void setZero(int** Ary){
       for (int i = 0; i < (numRows + 2); i++){
          for (int j = 0; j < (numCols + 2); j++){}
             Ary[i][j] = 0;
          }
       }
     }
     void loadImage(ifstream& inFile, int** ZFAry){
       for(int i = 1; i < (numRows + 1); i++){
          for (int j = 1; j < (numCols + 1); j++){
             inFile >> ZFAry[i][i];
          }
       }
     void prettyPrint(int** ZFAry, ofstream& prettyPrintFile){
       for (int i = 0; i < (numRows + 2); i++){
          for (int j = 0; j < (numCols + 2); j++){
             prettyPrintFile << ZFAry[i][j] << " ";</pre>
          prettyPrintFile << "\n";</pre>
       }
        prettyPrintFile << "\n";
     }
     void distanceTransform(int** ZFAry, int distanceChoice, ofstream& prettyPrintFile, ofstream& logFile){
        logFile << "Entering DistanceTransform() method";</pre>
       logFile << "\n";
       distancePass1(ZFAry, distanceChoice, logFile);
        prettyPrintFile << "1st pass distance transform with choice = " << distanceChoice << "\n";
```

```
prettyPrint(ZFAry, prettyPrintFile);
        distancePass2(ZFAry, distanceChoice, logFile);
        prettyPrintFile << "2nd pass distance transform with choice = " << distanceChoice << "\n";
        prettyPrint(ZFAry, prettyPrintFile);
        logFile << "Leaving DistanceTransform() method";</pre>
        logFile << "\n";
     }
     void distancePass1(int** ZFAry, int distanceChoice, ofstream& logFile){
        for(int i = 0; i < (numRows + 2); i++){
          for (int j = 0; j < (numCols + 2); j++){
             if(ZFAry[i][j] > 0){
                switch (distanceChoice)
                {
                case 8:
                   ZFAry[i][j] = (min(min(ZFAry[i-1][j-1], ZFAry[i-1][j]), min(ZFAry[i-1][j+1], ZFAry[i][j-1])) + 1);
                   break;
                case 4:
                   ZFAry[i][j] = (min(min(ZFAry[i-1][j-1]+2, ZFAry[i-1][j]+1), min(ZFAry[i-1][j+1]+2,
ZFAry[i][j-1]+1)));
                   break;
                default:
                   break;
                }
             }
          }
     }
     void distancePass2(int** ZFAry, int distanceChoice, ofstream& logFile){
        for(int i = (numRows + 1); i > 0; i - ){
           for (int j = (numCols + 1); j > 0; j--){
             if(ZFAry[i][i] > 0){
                switch (distanceChoice)
                case 8:
                   ZFAry[i][j] = min(min(min(ZFAry[i][j], ZFAry[i][j + 1] + 1), ZFAry[i + 1][j - 1] + 1), min(ZFAry[i][j] + 1] + 1)
+ 1][j] + 1, ZFAry[i + 1][j + 1] + 1));
                   break;
                case 4:
```

```
ZFAry[i][j] = min(min(min(ZFAry[i][j], ZFAry[i][j + 1] + 1), ZFAry[i + 1][j - 1] + 2), min(ZFAry[i][j] + 1] + 1)
+ 1][j] + 1, ZFAry[i + 1][j + 1] + 2));
                  break;
                default:
                  break;
                }
             }
          }
       }
     void compression(int** ZFAry, int distanceChoice, ofstream& skeletonFile, ofstream& prettyPrintFile,
ofstream& logFile){
       logFile << "Entering compression() method.";</pre>
       logFile << "\n";
       computeLocalMaxima(ZFAry, skeletonAry, distanceChoice, logFile);
        prettyPrintFile << "Local maxima, skeletonAry with choice = " << distanceChoice;</pre>
        prettyPrintFile << "\n";</pre>
        prettyPrint(skeletonAry, prettyPrintFile);
        extractSkeleton(skeletonAry, skeletonFile, logFile);
       logFile << "In compression() Below is skeleton Array with choice = " << distanceChoice;
       logFile << "\n";
        prettyPrint(skeletonAry, logFile);
       logFile << "Leaving compression() method";</pre>
       logFile << "\n";
     }
     void computeLocalMaxima(int** ZFAry, int** skeletonAry, int distanceChoice, ofstream& logFile){
       for(int i = 1; i < (numRows + 1); i++){}
          for (int j = 1; j < (numCols + 1); j++){
             if(isLocalMaxima(ZFAry, i, j)){
                skeletonAry[i][j] = ZFAry[i][j];
             }
             else{
                skeletonAry[i][j] = 0;
          }
       }
```

```
bool isLocalMaxima(int** ZFAry, int i, int j) {
  int currentVal = ZFAry[i][j];
  return (currentVal >= ZFAry[i-1][j-1] &&
        currentVal >= ZFAry[i-1][j] &&
        currentVal >= ZFAry[i-1][j+1] &&
        currentVal >= ZFAry[i][j-1] &&
        currentVal >= ZFAry[i][j+1] &&
        currentVal >= ZFAry[i+1][j-1] &&
        currentVal >= ZFAry[i+1][j] &&
        currentVal >= ZFAry[i+1][j+1]);
}
void extractSkeleton(int** skeletonAry, ofstream& skeletonFile, ofstream& logFile){
   skeletonFile << numRows << " " << numCols << " " << minVal << " " << maxVal;
  skeletonFile << "\n";
  for(int i = 0; i < (numRows + 2); i++){
     for(int j = 0; j < (numCols + 2); j++){}
        if(skeletonAry[i][j] > 0) {
          skeletonFile << i << " ";
          skeletonFile << j << " ";
          skeletonFile << skeletonAry[i][j];
          skeletonFile << "\n";
       }
     }
void loadSkeleton(ifstream& skeletonFile, int** ZFAry, ofstream& logFIle){
  int i;
  int j;
  int val;
  string line;
  getline(skeletonFile, line);
  while(getline(skeletonFile, line)){
     istringstream iss(line);
     iss >> i;
     iss >> j;
     iss >> val;
     ZFAry[i][j] = val;
  }
}
```

```
void deCompression(int** ZFAry, int distanceChoice, ofstream& prettyPrintFile, ofstream& logFile){
  logFile << "Entering deCompression() method";</pre>
  logFile << "\n";
  expansionPass1(ZFAry, distanceChoice, logFile);
   prettyPrintFile << "1st pass expansion with choice = " << distanceChoice;</pre>
   prettyPrintFile << "\n";
  prettyPrint(ZFAry, prettyPrintFile);
  expansionPass2(ZFAry, distanceChoice, logFile);
   prettyPrintFile << "2nd pass expansion with choice = " << distanceChoice;</pre>
  prettyPrintFile << "\n";
  prettyPrint(ZFAry, prettyPrintFile);
}
void expansionPass1(int** ZFAry, int distanceChoice, ofstream& logFile){
  for(int i = 1; i < (numRows + 1); i++){
     for(int j = 1; j < (numCols + 1); j++){
        if(ZFAry[i][j] == 0){
          int maxVal = ZFAry[i][j];
          switch (distanceChoice)
          {
          case 8:
             for (int row = i - 1; row <= i + 1; row++) {
                for (int col = j - 1; col <= j + 1; col++) {
                  maxVal = max(maxVal, ZFAry[row][col]-1);
               }
             ZFAry[i][j] = maxVal;
             break;
          case 4:
             ZFAry[i][j] = max(max(max(max(max(ZFAry[i][j], ZFAry[i][j - 1]-1),
             ZFAry[i][j + 1]-1),
             ZFAry[i - 1][j - 1]-2),
             ZFAry[i - 1][j]-1),
             max(ZFAry[i - 1][j + 1]-2,
             max(ZFAry[i + 1][j - 1]-2,
             max(ZFAry[i + 1][j]-1,
             ZFAry[i + 1][j + 1]-2))));
             break;
          default:
             break;
```

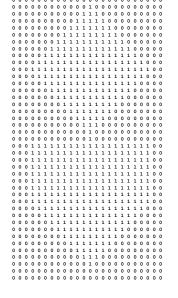
```
}
       }
    }
  }
}
void expansionPass2(int** ZFAry, int distanceChoice, ofstream& logFile){
  for(int i = (numRows); i > 0; i--){
     for(int j = (numCols); j > 0; j--){
        int maxVal = ZFAry[i][j];
        switch (distanceChoice)
        {
        case 8:
          for (int row = i - 1; row <= i + 1; row++) {
             for (int col = j - 1; col <= j + 1; col++) {
                maxVal = max(maxVal, ZFAry[row][col]-1);
             }
          }
          ZFAry[i][j] = maxVal;
        break;
        case 4:
          ZFAry[i][j] = max(max(max(max(max(ZFAry[i][j], ZFAry[i][j - 1]-1),
          ZFAry[i][j + 1]-1),
          ZFAry[i - 1][j - 1]-2),
          ZFAry[i - 1][j]-1),
          max(ZFAry[i - 1][j + 1]-2,
          max(ZFAry[i + 1][j - 1]-2,
          max(ZFAry[i + 1][j]-1,
          ZFAry[i + 1][j + 1]-2))));
        break;
        default:
          break;
        }
     }
  }
}
void binThreshold(int** ZFAry, ofstream& deCompressedFile){
  for(int i = 0; i < (numRows + 2); i++){
     for(int j = 0; j < (numCols + 2); j++){
        if(ZFAry[i][j] >= 1){
          deCompressedFile << "1 ";
        else{
```

```
deCompressedFile << "0 ";
             }
          }
          deCompressedFile << "\n";
       }
     }
};
int main(int argc, char** argv){
  //Creates an instance of each file passed as an argument and makes sure they can be opened.
  ifstream inFile(argv[1]);
  if(!inFile.is_open()){
     cout << "Unable to open: " << argv[1];
     exit(1);
  }
  ofstream prettyPrintFile(argv[3]);
  if(!prettyPrintFile.is_open()){
     cout << "Unable to open: " << argv[3];
     exit(1);
  }
  ofstream skeletonFile(argv[4]);
  if(!skeletonFile.is_open()){
     cout << "Unable to open: " << argv[4];
     exit(1);
  }
  ofstream deCompressedFile(argv[5]);
  if(!deCompressedFile.is_open()){
     cout << "Unable to open: " << argv[5];
     exit(1);
  }
  ofstream logFile(argv[6]);
  if(!logFile.is open()){
     cout << "Unable to open: " << argv[6];
     exit(1);
  }
```

```
//Creates an instance of the distance skeleton class.
  DistanceSkeleton* distanceSkeleton = new DistanceSkeleton;
  //Extracts the numRows, numCols, minVal, and maxVal values from the input file.
  inFile >> distanceSkeleton->numRows:
  inFile >> distanceSkeleton->numCols;
  inFile >> distanceSkeleton->minVal;
  inFile >> distanceSkeleton->maxVal:
  //Dynamically allocates ZFAry and skeletonAry to have 2 extra rows and columns.
  distanceSkeleton->ZFAry = new int*[distanceSkeleton->numRows + 2];
  distanceSkeleton->skeletonAry = new int*[distanceSkeleton->numRows + 2];
  for(int i = 0; i < distanceSkeleton->numRows + 2; i++){
     distanceSkeleton->ZFAry[i] = new int[distanceSkeleton->numCols + 2];
    distanceSkeleton->skeletonAry[i] = new int[distanceSkeleton->numCols + 2];
  }
  //Sets distanceChoice to the second argument passed.
  distanceSkeleton->distanceChoice = atoi(argv[2]);
  //Sets al values of ZFAry and skeletonAry to zero.
  distanceSkeleton->setZero(distanceSkeleton->ZFAry);
  distanceSkeleton->setZero(distanceSkeleton->skeletonAry);
  //Loads the image from the inFile into the ZFAry
  distanceSkeleton->loadImage(inFile, distanceSkeleton->ZFAry);
  distanceSkeleton->prettyPrint(distanceSkeleton->ZFAry, prettyPrintFile);
  distanceSkeleton->distanceTransform(distanceSkeleton->ZFAry, distanceSkeleton->distanceChoice,
prettyPrintFile, logFile);
  distanceSkeleton->compression(distanceSkeleton->ZFAry, distanceSkeleton->distanceChoice,
skeletonFile, prettyPrintFile, logFile);
  skeletonFile.close();
  ifstream skeletonInputFile(argv[4]);
  if(!skeletonInputFile.is_open()){
    cout << "Unable to open: " << argv[4] << " for reading";
     exit(1);
  }
```

```
distanceSkeleton->setZero(distanceSkeleton->ZFAry);
  distanceSkeleton->loadSkeleton(skeletonInputFile, distanceSkeleton->ZFAry, logFile);
  prettyPrintFile << "Below is the loaded skeleton with choice = " << distanceSkeleton->distanceChoice;
  prettyPrintFile << "\n";
  distanceSkeleton->prettyPrint(distanceSkeleton->ZFAry, prettyPrintFile);
  distanceSkeleton->deCompression(distanceSkeleton->ZFAry, distanceSkeleton->distanceChoice,
prettyPrintFile, logFile);
  deCompressedFile << distanceSkeleton->numRows << " " << distanceSkeleton->numCols << " " <<
distanceSkeleton->minVal << " " << distanceSkeleton->maxVal;
  deCompressedFile << "\n";
  distanceSkeleton->binThreshold(distanceSkeleton->ZFAry, deCompressedFile);
  skeletonInputFile.close();
  inFile.close();
  prettyPrintFile.close();
  deCompressedFile.close();
  logFile.close();
}
```

PrettyPrint for img1 using 4-distance



1st	pas	s	iis	sta	ano	e	tı	car	ns:	£01	cm	w	t	h e	cho	oic	e	=	4			
0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	0 0																					
	0 0																					
0 0		0																		0	0	
	0 0																			0	0	
0 0		0														0					0	
0 0		0																0			0	
0 0		0																				
	0 0																					
0 0		1																				
0 0		. 2																				
0 0		1																				
0 0		0																				
0 0		0																				
	0 0																					
0 0		0																				
	0 0																					
	0 0																		0			
0 0		0																0			0	
0 0		0																			0	
	0 0																			0		
0 0		. 2																		0		
	0 1																					
	0 1																		2			
0 0		2																				
	0 1																					
	0 1																					
	0 1																					
0 0		2																				
0 0																					0 0	n
0 0																					0 0 0	
0 0	0 0	0	0	1	2	3	4	5	6	7	8	9	10	0 :	11	10) 8	3 (0 0		0 0	
0 0																					0 0	
0 0	0 0	0	0	0	0	1	2	3	4	5	6	7	8	9	0	0	0	o	0	0	0	
0 0		0																				
0 0		0																	0			
0 0	0 0	0	0	0	0	0	0	0	1	2	3	0	0	0	0	0	0	0	0	0	0	
0 0	0 0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	
0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0 0	0 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
										_												

Lo	oce	1	ma	ax:	im	a,	sl	ce:	Let	tor	nA:	сy	w	L th		che	oio	ce	-	4			
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	10	0	0) () (0 (0	0) (0 (0	0
0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	-	-	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0		-	່	-	-	-	-	່	-	-	-	່ເ
0	0	o	0	0	0	0	0	0	0	0	0	-			0				0			0	0
ō	0	o	0	ō	0	0	0	ō	ō	0	0	o	ō	ō	o	o	0	ō	ō	o	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	ō	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

18	зŧ	ра	18	3 6	ext	oar	ns	Los	2 1	vi1	th	cl	201	Lae			4						
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	o	0	0	0	o	0	0	0	o	0	0	0
0	0	0	0	0	0	0	0	0	0	0	8	9	8	-	6	ŗ	•	3	۰	1	٥	~	۰
	0				0						9			٠.	٠.	٠.	4.	. 3	٠.	٠.	Ü.	٠,	Ů.
0		0	0	0		0	0	0	0	6		10		-	5 .	′ '	ь:	٠.		3 .	۷.	٠. ١	٠.
0	0	0	0	0	0	0	0	0	5	7 6	8 7	9	8 7	7	6 5	5 4	4	2	2	1	0	0	0
0																							
0	0	0	0	0	0	0	0	3	4	5	6	7	6	5	4	3	2	1	0	0	0	0	0
0	0	0	0	0	0	0	1	2	3	4	5	6	5	4	3	2	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	2	3	4	5	4	3	2	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	2	3	4	3	2	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	2	3	2	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	2	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
'n	0	0	0	0	0	0	0	0	0	0	8	9	8	7	6	5	4	3	2	1	0	n	'n
0	0	0	0	0	0	0	0	0	0	6	9	10	ň	,	ŭ,	, ,			ī.	٠,	, ĭ.	ŭ	ň
0	0	0	0	0	0	0	0	0	4	7	8	9	8	7	6	5	4	3	2	1	0	0	0
0	0	0	0	0	0	0	0	2	5	6	7	8	7	6	5	4	3	2	1	0	0	0	0
0	0	0	0	0	0	0	0	3	4	5	6	7	6	5	4	3	2	1	0	0	0	0	0
0	0	0	0	0	0	0	1	2	3	4	5	6	5	4	3	2	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	2	3		5	4	3	2	1	0	0	0	0	0	0	0
			0		0				1	2	4		3		2	1	0	0	0	0		0	0
0	0	0		0		0	0	0			2	4		2	1 0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0		0	1			2	1			0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	2	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0																						
0	U	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	-	-	0	0	0	-	-	Ī	Ī	0	-	-	0	0	0	0	0	0	0	0	0	0	0
2:	nd	pi	ası	3 6	×	oar	ns:	Los	2 1	vi1	th	cl	10:	Loe			4		Ĭ		•	•	0
2:	nd 0	pa 0	0	0	ex;	oar 0	0	Los 0	0	vi1	th 0	c1 0	0	0	0		4	0	0	0	0	0	0
0 2: 0 0	nd 0 0	pa 0	0	0	0 0	0 0	0	0	0	vi1	0 0	0	0	0	0	0	0 0	0	0	0	0	0	0
2: 0 0	0 0 0	pa 0 0	0 0	0 0	0 0	0 0 0	0	0 0	0 0	vi1 0 0	0 0 0	0 0	0 0	0 0	0 0	000	0 0	0 0	0 0 0	0 0	0 0	0 0	0
2: 0	nd 0 0	pa 0	0	0	0 0	0 0 0 0	0	0 0	0	0 0 0 0	0 0 0	0 0 1 2	0 0 0	0 0	0 0 0	0000	0 0 0	0 0 0	0	0 0 0	0 0 0	0 0 0	0
2: 0 0	0 0 0	pa 0 0	0 0	0 0	0 0	0 0 0	0	0 0 0	0 0	vi1 0 0	0 0 0	0 0	0 0 1 2	0 0	0 0 0 0	00000	0 0	0 0	0 0 0	0 0	0 0	0 0 0 0	0
2: 0 0 0	0 0 0	pi 0 0 0	0 0	0 0 0	0 0	0 0 0 0	0 0	0 0	0 0	0 0 0 0	0 0 0	0 0 1 2	0 0 0	0 0	0 0 0	0000	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0	0
2: 0 0 0 0	0 0 0 0 0	pi 0 0 0 0	0 0 0 0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0 0 1	vii 0 0 0 0 1 2	0 0 1 2	0 0 1 2	0 0 0 1 2 3	0 0 0 0 1 2	0 0 0 0 0 1	00000	0 0 0 0	0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 0
21 0 0 0 0 0	0 0 0 0	pa 0 0 0 0 0 0 0	0 0 0	0 0 0	0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0	vii 0 0 0 0 1 2 3	0 0 0 1 2	0 0 1 2 3	0 0 1 2	0 0 0 1	0 0 0 0	00000	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0 0	0 0 0 0	0 0 0
21 0 0 0 0 0	nd 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 1 2	0 0 0 0 1 2 3	vii 0 0 0 0 1 2	th 0 0 1 2 3 4 5	0 0 1 2 3 4 5	0 0 0 1 2 3 4 5	0 0 0 0 1 2	0 0 0 0 1 2 3	00000012	4 0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0
21 0 0 0 0 0 0 0	nd 0 0 0 0 0 0 0 0 0	P8 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 1 2	0 0 0 0 1 2 3	0 0 0 0 1 2 3 4	vii 0 0 0 0 1 2 3	th 0 0 1 2 3 4 5 6	0 0 1 2 3 4 5 6 7	0 0 0 1 2 3	0 0 0 0 1 2 3	0 0 0 0 1 2 3 4	0 0 0 0 0 1 2 3	1 0 0 0 0 0 0 0 1 2	0 0 0 0 0 0 1	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0
2: 0 0 0 0 0 0 0	nd 0 0 0 0 0 0 0 0	pi 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 1 2	0 0 0 0 1 2 3 4 5	vii 0 0 0 0 1 2 3	th 0 0 0 1 2 3 4 5 6 7	cl 0 0 1 2 3 4 5	0 0 0 1 2 3 4 5	0 0 0 1 2 3 4	0 0 0 0 1 2 3 4 5	00000012	1 0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0
2: 0 0 0 0 0 0 0 0 0	nd 0 0 0 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1	0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 2 3	0 0 0 0 0 0 0 1 2 3	0 0 0 0 1 2 3 4 5	0 0 0 0 1 2 3 4 5 6	vii 0 0 0 0 1 2 3 4 5 6 7	th 0 0 1 2 3 4 5 6 7 8	cl 0 0 1 2 3 4 5 6 7 8 9	0 0 0 1 2 3 4 5 6 7 8	0 0 0 0 1 2 3 4 5	0 0 0 0 1 2 3 4	0 0 0 0 0 1 2 3	1 0 0 0 0 0 0 0 1 2	0 0 0 0 0 0 1	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0
2: 0 0 0 0 0 0 0 0 0	ad 0 0 0 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1	00000000012	0 0 0 0 0 0 0 0 0 0 1 2 3	0 0 0 0 0 1 2 3 4	0 0 0 0 0 1 2 3 4 5	0 0 0 0 0 1 2 3 4 5 6	0 0 0 0 1 2 3 4 5 6 7	vii 0 0 0 0 1 2 3 4 5 6 7 8	th 0 0 0 1 2 3 4 5 6 7 8 9	cl 0 0 1 2 3 4 5 6 7 8 9 10	0 0 0 1 2 3 4 5 6 7 8	0 0 0 1 2 3 4 5 6 7	0 0 0 0 0 1 2 3 4 5 6	0000012345	4 0 0 0 0 0 0 1 2 3 4 6 !	0 0 0 0 0 0 1 2 3	0 0 0 0 0 0 0 1 2	0 0 0 0 0 0 0 0 1 3	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0	P8 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 0	0 0 0 0 0 0 0 1 2 1	0 0 0 0 0 0 0 0 0 1 2 3	0 0 0 0 0 1 2 3 4 3	0 0 0 0 0 1 2 3 4 5 4	0 0 0 0 0 1 2 3 4 5 6 5	0 0 0 0 1 2 3 4 5 6 7 6	vii 0 0 0 0 1 2 3 4 5 6 7 8 7	th 0 0 0 1 2 3 4 5 6 7 8 9 8	cl 0 0 1 2 3 4 5 6 7 8 9 10 9	0 0 0 1 2 3 4 5 6 7 8 9 8	0 0 0 1 2 3 4 5 6 7	0 0 0 0 0 1 2 3 4 5 6 3 6	0000012345	4 0 0 0 0 0 0 1 2 3 4 5 4	0 0 0 0 0 0 1 2 3 5 4	0 0 0 0 0 0 1 2 1 2	0 0 0 0 0 0 0 0 1 3 1	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 L 0	000000000
2:0000000000000000000000000000000000000	000000000000000000000000000000000000000	P ² 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 0 0	0 0 0 0 0 0 0 1 2 1 0	0 0 0 0 0 0 1 2 3 2 1	0 0 0 0 0 1 2 3 4 3 2	0 0 0 0 0 1 2 3 4 5 4 3	0 0 0 0 1 2 3 4 5 6 5 4	0 0 0 0 1 2 3 4 5 6 7 6 5	vii 0 0 0 0 1 2 3 4 5 6 7 8 7 6	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7	0 0 1 2 3 4 5 6 7 8 9 10 9 8	0 0 0 1 2 3 4 5 6 7 8 9 8 7	0 0 0 1 2 3 4 5 6 7 6	0 0 0 0 0 1 2 3 4 5 6 5	0000012345	1 0 0 0 0 0 0 1 2 3 4 3	0 0 0 0 0 0 1 2 3 5 3 2	0 0 0 0 0 0 1 2 1 2 1	0 0 0 0 0 0 0 0 1 3 1 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0 0 0	P ² 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 0 0 0	0 0 0 0 0 0 0 1 2 1 0 0	0 0 0 0 0 0 1 2 3 2 1 0	0 0 0 0 0 1 2 3 4 3 2 1	0 0 0 0 0 1 2 3 4 5 4 3 2	0 0 0 0 0 1 2 3 4 5 6 5 4 3	000001234567654	vii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6	cl 0 0 1 2 3 4 5 6 7 8 9 10 9 8 7	0 0 0 1 2 3 4 5 6 7 8 9 8 7 6	0 0 0 0 1 2 3 4 5 6 7 6 5	0 0 0 0 0 1 2 3 4 5 6 5	0000012345	1 0 0 0 0 0 0 1 2 3 4 3 2	0 0 0 0 0 0 1 2 3 5 4 3 2 1	0000000121210	0 0 0 0 0 0 0 1 3 1 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	3 0 0 0 0 0 0 0 0 1 2 1 0 0 0	0 0 0 0 0 0 1 2 3 2 1 0 0	0 0 0 0 0 0 1 2 3 4 3 2 1 0	0 0 0 0 0 0 1 2 3 4 5 4 3 2 1	0000001234565432	0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3	vii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	cl 0 0 1 2 3 4 5 6 7 8 9 10 9 8 7 6	0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4	0 0 0 0 0 1 2 3 4 5 6 5 4 3	000001234575432	4 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1	0 0 0 0 0 0 1 2 3 5 3 2 1 0	0 0 0 0 0 0 0 1 2 1 2 1 0 0	0 0 0 0 0 0 0 0 1 3 1 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	00000000000000
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 2 1 0 0 0 0	0 0 0 0 0 0 1 2 3 2 1 0 0 0	0 0 0 0 0 0 1 2 3 4 3 2 1 0 0	0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0	00000012345654321	0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3 2	vii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	c1 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5	0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3	00000123456365432	0000012345	4 0 0 0 0 0 0 0 1 2 3 4 9 4 3 2 1 0	0 0 0 0 0 0 0 1 2 3 5 3 2 1 0 0	0 0 0 0 0 0 0 1 2 1 0 0 0	0 0 0 0 0 0 0 0 1 3 1 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000	00000000000000000
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000001210000	0 0 0 0 0 0 1 2 3 2 1 0 0 0	0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0	00000012345432100	000000123456543210	000001234567654321	vi1000012345678765432	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3	c1 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5	0000123456789876543	0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2	000012345654321	00000012345,543210	1 0 0 0 0 0 0 1 2 3 4 9 4 3 2 1 0 0	0 0 0 0 0 0 0 1 2 3 5 3 2 1 0 0 0	0 0 0 0 0 0 0 0 1 2 1 0 0 0 0	000000001	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	400000000000000000000000000000000000000	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	0000000012100000	0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0	0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0	000000123454321000	0000001234565432100	0000012345676543210	vi1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1	h0001234567898765432	c1 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3	0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2	0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1	00000123456543210	0000001234575432100	1 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0	0 0 0 0 0 0 0 1 2 3 5 3 2 1 0 0 0 0	00000001212100000	0 0 0 0 0 0 0 0 1 3 1 0 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	100000000000000000000000000000000000000	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	00000000121000000	0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0 0	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0	0000001234543210000	00000012345654321000	00000123456765432100	vii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1	c1 0 0 1 2 3 4 5 6 7 8 9 10 9 8 7 6 5 4 3 2	0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1	000012345676543210	00000123456365432100	0000012345754321000	4 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0	0000000123532100000	00000001212100000	00000000131000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	200000000000000000000000000000000000000	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	0000000001210000000	000000012321000000	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0	0000001234543210000	000000123456543210000	000001234567654321000	000012345678765432100	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0	c1001234567891987654321	0000123456789876543210	000012345678765432100	0000012345654321000	000000123457543210000	1 0 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0	0000000123532100000	0000000012121000000	000000001	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	200000000000000000000000000000000000000	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	0 0 0 0 0 0 0 0 0 1 2 1 0 0 0 0 0 0 0 0	000000012321000000	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0	0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0	000000123456543210000	0000012345676543210000	0000123456787654321000	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0	c1 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1	0001234567898765432100	0000123456787654321000	0000012345636543210000	0000012345 5432100000	1 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0	00000001235321000000	00000000121210000000	0000000013100000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	100000000000000000000000000000000000000	200000000000000000000000000000000000000	000000000000000000000000000000000000000	3 0 0 0 0 0 0 0 0 0 1 2 1 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0	000000123454321000000	0000001234565432100000	100000123456765432100000	00001234567876543210000	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1	c100123456789198765432112	00012345678987654321001	00001234567876543210000	00000123456365432100000	00000012345754321000000	1 0 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0	000000012353210000000	000000001212100000000	00000000131000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	100000000000000000000000000000000000000	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	3 0 0 0 0 0 0 0 0 0 1 2 1 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0	0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0	00000012345654321000000	100000123456765432100000	·i1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0 0 0 0 1	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2	cl 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1 2 3	0000123456789876543210012	000012345678765432100001	000001234563654321000000	00000012345,54321000000	1 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0	0000000123532100000000	000000012121000000000	00000000013100000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	00000000121000000000000	0000000123210000000000	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0	000000123456543210000000	1000001234567654321000001	·i1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0 0 0 0 1 2	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2 3	010012345678919876543211234	0 0 0 0 1 2 3 4 5 6 7 8 8 7 6 5 4 3 2 1 0 0 1 2 3	0000123456787654321000012	0000012345636543210000001	00000012345,5432100000000	1 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0	00000001235321000000000	000000001212100000000000	00000000131000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	ad 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P. 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	000000001210000000000000	00000001232100000000000	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0	000000123456543210000000	10000012345676543210000012	vi100001234567876543210000123	h 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2 3	0100123456789198765432112345	00001234567858765432100123	000012345678765432100001	0000012345636543210000012	00000012345 5432100000001	4 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0	00000012353210000000000	00000001212100000000000	00000000013100000000000000	000000000000000000000000000000000000000	00000000000000000000000000000000000000	000000000000000000000000000000000000000
2:0000000000000000000000000000000000000	100000000000000000000000000000000000000	P0000000000000000000000000000000000000	000000000000000000000000000000000000000	0000000012100000000000000	0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0 0 0 0	000000123454321000000001	00000012345654321000000012	100000123456765432100000123	11000012345678765432100001234	h000123456789876543210012345	cl 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6	0000123456789876543210012345	000012345678765432100001234	00000123456365432100000123	00000012345 54321000000012	4 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0 1	0000000123532100000000000	00000000121210000000000000	000000000131000000000000000	000000000000000000000000000000000000000	000000000 C00000000000000	000000000000000000000000000000000000000
220000000000000000000000000000000000000	100000000000000000000000000000000000000	P0000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000121000000000000000	0 0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0 0 0 0 1	0000001234543210000000012	0000000123456543210000000123	1000001234567654321000001234	110000123456787654321000012345	h000123456789876543210012345	cl 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7	000012345678 8765432100123456	0000123456787654321000012345	000001234563654321000001234	00000012345 543210000000123	4 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0 0 1 2	00000001235321000000000000	000000001212100000000000000	0000000001310000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2r 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	400000000000000000000000000000000000000	P0000000000000000000000000000000000000	000000000000000000000000000000000000000	00000000012100000000000000000	0 0 0 0 0 0 0 0 1 2 3 2 1 0 0 0 0 0 0 0 0 0 0 1	0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0 0 0 1 2	00000012345432100000000123	0000001234565432100000001234	0000012345676543210000012345	·i100001234567876543210000123456	h00012345678987654321001234567	cl 0 0 0 1 2 3 4 5 6 7 8 9 1 0 9 8 7 6 5 4 3 2 1 1 1 2 3 4 5 6 7 8	000012345678 87654321001234567	000012345678765432100001234	0000012345636543210000012345	00000012345 54321000000012	4 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0 1	00000001235321000000000012	00000000121210000000000000	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2m 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	400000000000000000000000000000000000000	P0000000000000000000000000000000000000	000000000000000000000000000000000000000	00000000012100000000000000000	00000001232100000000000012	ar 0 0 0 0 0 0 0 1 2 3 4 3 2 1 0 0 0 0 0 0 0 0 0 0 1 2 3	000000123454321000000001234	00000012345654321000000012345	00000123456765432100000123456	·i1000012345678765432100001234567	h000123456789876543210012345678	cl 0 0 0 1 2 3 4 5 6 7 8 9 1 0 9 8 7 6 5 4 3 2 1 1 1 2 3 4 5 6 7 8 9	000012345678 876543210012345678	0000123456787654321000012345	000001234563654321000001234	00000012345 543210000000123	4 0 0 0 0 0 0 1 2 3 4 5 4 3 2 1 0 0 0 0 0 0 0 0 0 1 2	00000001235321000000000000	000000001212100000000000000	0000000001310000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000	000000000000000000000000000000000000000
2n 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	d 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	120000000000000000000000000000000000000	000000000000000000000000000000000000000	00000000012100000000000000012	00000000123210000000000000123	0000000123432100000000001234	00000012345432100000000012345	000000123456543210000000123456	10000012345676543210000001234567	110000123456787654321000012345678	h0001234567898765432100123456789	cl 0 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1 1 2 3 4 5 6 7 8 9 1 0	000012345678 876543210012345678	00001234567 765432100001234567	000001234563654321000001234563	00000012345755432100000012334557	4 0 0 0 0 0 0 1 2 3 4 ! 4 3 2 1 0 0 0 0 0 0 0 0 1 2 3 4 !	0000000123532100000000001235	0000000012121000000000000121	000000000000000000000000000000000000000	000000000000000000000000000000000000000	00000000000000000000000000000000000000	000000000000000000000000000000000000000
2m	and 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dear 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ns: 0 0 0 0 0 0 0 1 2 3 4 4 5 5 4 3 2 2 1 0 0 0 0 0 0 0 1 2 2 3 4 4 5 4 4 5 4 5 4 5 4 5 4 5 4 5 6 6 6 6	100 00 00 00 00 00 12 34 56 55 43 22 10 00 00 00 00 12 33 45 65 65 65 65 65 65 65 65 65 65 65 65 65	2 0 0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3 2 1 0 0 0 0 0 0 1 2 3 4 5 6 7 6	vit 0 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 0 0 0 0 1 2 3 4 5 6 7 8 7	hh 0 0 0 1 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2 3 4 5 6 7 8 9 8	cl 0 0 0 1 2 3 4 5 6 7 8 9 9 1 9 8 7 6 5 4 3 2 2 1 1 2 3 4 5 6 7 8 9 1 9	00012345678998765432100112345678998	1 C C C C C C C C C C C C C C C C C C C	" " 0 0 0 0 0 0 1 2 3 3 4 5 5 6 5 5 4 3 2 2 1 0 0 0 0 0 1 2 3 3 4 5 5 6 6 5 6 5 6 6 7 6 7 6 7 6 7 6 7 6 7	000000123345575432210000001223445575	4 0 0 0 0 0 0 0 0 1 2 3 4 4 1 3 2 2 1 0 0 0 0 0 0 0 1 2 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00000000000000000000000000000000000000	
2m	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dear 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	100 00 00 00 00 00 12 12 33 44 55 44 32 21 10 00 00 00 00 00 00 00 00 00 00 00 00	10000012345676543210000001234567	wii 1 0 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 1 1 0 0 0 0 1 2 3 4 5 6 7 8 7 6	hh 0 0 0 1 2 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2 3 4 5 6 7 8 9 8 7	cl 0 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1 1 2 3 4 5 6 7 8 9 1 0	000000000000000000000000000000000000000	00001234567 765432100001234567	" 0 0 0 0 0 0 0 1 1 2 2 3 3 4 5 5 6 5 5 6 5 5 6 5 5 6 5 5 6 5 5 6	0000001233457543221000000123345754	4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00000000000000000000000000000000000000	
2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P** 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dear 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	100 0000000000000000000000000000000000	0 0 0 0 0 0 1 2 2 3 4 4 5 6 7 6 5 4 3 2 2 1 0 0 0 0 0 1 2 2 3 4 4 5 6 7 6 5 4	wii 1 0 0 0 0 0 1 2 3 3 4 5 6 7 8 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5	hth 0 0 0 1 2 2 3 4 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 1 2 3 4 4 5 6 7 8 9 8 7 6	cl 0 0 1 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7 8 9 1 9 8 7	000000000000000000000000000000000000000	0 0 0 0 0 1 2 2 3 4 4 5 5 6 7 7 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 3 4 5 5 6 7 7 6 5 5	" 0 0 0 0 0 0 1 1 2 2 3 4 4 5 6 6 5 4 3 2 2 1 0 0 0 0 0 1 2 2 3 4 4 5 6 6 5 6 5 6 5 6 5 6 5 6 5 6 5 6 5	00000012344575443221000000112344575433	4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	00000000000000000000000000000000000000	
2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	and 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dark 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 1 2 3 3 4 5 5 4 3 3 2 2 1 0 0 0 0 0 0 0 1 2 3 3 4 5 5 4 3 3 2 1 1	1001 000000000000000000000000000000000	0 0 0 0 0 0 1 2 3 4 4 5 6 6 7 6 6 5 4 3 2 2 1 0 0 0 0 0 0 1 2 3 4 5 6 6 7 6 5 4 3	viii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	cl 0 0 1 1 2 3 4 5 6 7 8 9 1 1 9 8 7 6 5 4 3 2 1 1 2 3 4 5 6 7 8 9 1 9 8 7 6	0 0 0 0 1 2 3 4 5 6 6 7 8 8 9 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 5 6 7 8 8 7 6 5 5 6 7 8 9 8 7 6 5	1000000123445667654322100001123445667654	0 0 0 0 0 0 1 2 3 3 4 5 5 6 5 5 4 4 3 2 2 1 0 0 0 0 0 0 1 2 3 3 4 5 5 6 5 4 3 3 6 5 5 4 3	00000012345754432210000001233455754432	4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	and 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Description 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1001 000000000000000000000000000000000	0 0 0 0 0 0 1 2 2 3 4 4 5 6 7 6 5 4 3 2 2 1 0 0 0 0 0 1 2 2 3 4 4 5 6 7 6 5 4	viii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	010012344567891098765432112344567891098765	0 0 0 0 1 2 3 4 5 6 6 7 8 8 9 6 5 4 4 3 2 2 1 0 0 0 1 2 3 4 4 5 6 6 7 8 8 7 6 5 5	1 icc 0 0 0 0 0 1 2 3 4 4 5 6 6 7 7 6 5 5 4 3 3 2 2 1 0 0 0 0 1 2 3 3 4 5 6 6 7 6 5 4 3	0 0 0 0 0 0 1 2 3 3 4 5 5 6 5 5 4 4 3 2 2 1 0 0 0 0 0 0 1 2 3 3 4 5 5 6 5 4 3 3 6 5 5 4 3	00000011233455754332110000000112334557543321	4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	axi 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	DATE OF THE OF T	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 to	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	viii 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 8 7 6 5 4	hth 0 0 0 1 2 2 3 4 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 1 2 3 4 4 5 6 7 8 9 8 7 6	010012345678910987654321123445678910987654	0 0 0 0 0 1 1 2 3 3 4 5 6 6 7 8 8 8 7 6 5 5 4 3 2 1 0 0 0 1 2 3 3 4 5 6 7 8 8 7 6 5 5 4 3	1000000123445667654322100001123445667654	0 0 0 0 0 0 1 1 2 3 3 4 5 5 6 5 4 3 3 2 1 1 0 0 0 0 0 0 1 1 2 3 3 4 5 6 6 5 4 3 2 1 1	0000001223445754322100000001223445575443210	4 0 0 0 0 0 0 0 0 0 1 2 2 3 4 4 3 3 2 2 1 0 0 0 0 0 0 0 1 2 3 3 4 4 3 3 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	and 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dari 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 2 2 3 4 4 5 4 4 3 2 2 1 1 0 0 0 0 0 0 0 1 2 2 3 4 4 5 4 4 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	10000000000000000000000000000000000000	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	viii 100000123455678765432100000123456787654321	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	010012344567891098765432112344567891098765	0 0 0 0 1 2 3 4 5 6 6 7 8 8 9 6 5 4 4 3 2 2 1 0 0 0 1 2 3 4 4 5 6 6 7 8 8 7 6 5 5	1000000123345667 76554322100000122345667 765543221	0 0 0 0 0 0 1 1 2 3 3 4 5 6 6 5 4 3 3 2 1 1 0 0 0 0 0 0 1 1 2 3 3 4 5 6 6 5 4 3 2 1 1 0	00000012344557544322100000001234455754322100	4 0 0 0 0 0 0 0 1 2 2 3 4 4 3 2 2 1 0 0 0 0 0 0 0 1 2 3 4 4 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000		
21 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dear 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	10000000000000000000000000000000000000	1 0 0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3 2 1 0 0	vii0000012345678765432100001234567876543210	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1	cl 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 2 1 1 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2	0 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 6 7 8 9 9 1 8 7 6 5 4 3 2 1	1 correction 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 1 0	0 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0	0000001234457543221000000012234557543221000	4 0 0 0 0 0 0 0 1 2 3 4 4 3 2 2 1 0 0 0 0 0 0 1 2 3 4 4 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000		
2: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	and 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	P4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	2 2 1 0 0 0 0 0 0 0 0 1 2 2 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dear 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 1 2 2 3 4 4 5 4 4 3 2 2 1 0 0 0 0 0 0 0 1 2 2 3 4 4 5 4 4 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 to	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	viii 100000123455678765432100000123456787654321	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5	cl0 0 0 1 2 2 3 4 5 6 7 8 9 1 0 9 8 7 6 5 4 3 2 2 1 1 2 3 4 5 6 7 8 9 1 0 9 8 7 6 5 4 3 2 1	000001234567887654322100123456788876543210	1 correction 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 1 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 1 0 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 1 0 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 1 2 3 4 5 6 5 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 5 4 3 2 2 1 0 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0 0 0	00000012344557544322100000001234455754322100	4 0 0 0 0 0 0 0 1 2 2 3 4 4 3 2 2 1 0 0 0 0 0 0 0 1 2 3 4 4 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000	
21 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Pi 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	Dear 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	10000000000000000000000000000000000000	1 0 0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 7 6 5 4 3 2 1 0 0	vii0000012345678765432100001234567876543210	th 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 1	cl 0 0 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2 2 1 1 1 2 3 4 5 6 7 8 9 1 9 8 7 6 5 4 3 2	0 0 0 0 1 2 3 4 5 6 7 8 9 8 7 6 5 4 3 2 2 1 0 0 0 1 2 3 4 5 6 7 8 9 9 1 8 7 6 5 4 3 2 1	1 correction 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 2 1 0 0 0 0 1 2 3 4 5 6 7 7 6 5 4 3 2 1 0	0 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0 0 0 0 1 2 3 4 5 6 6 5 4 3 2 2 1 0 0	0000001234457543221000000012234557543221000	4 0 0 0 0 0 0 0 1 2 3 4 4 3 2 2 1 0 0 0 0 0 0 1 2 3 4 4 3 2 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	000000000000000000000000000000000000000	000000000000000000000000000000000000000		

SkeletonFlle for img1 using 4-distance $^{\tiny 40\ 22\ 0\ 1}_{\tiny 11\ 12\ 10}_{\tiny 30\ 12\ 10}$

deCompressedFile for img1 using 4-distance

logFile for img1 using 4-distance

Entering DistanceTransform() method Leaving DistanceTransform() method

Leaving compression() method Entering deCompression() method

F)	re	e 1	tt	·	ı F	>	r	i	n	t	1	fo	וכ	r	i	n	n	Q	11	I	ι	using 8-distance	
0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0	0 0 0	0	0	0 0 0	0 0 0	0 0 0	0 0 1	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0 0 0 0	
0 0 0	0 0 0	0 0 0	0 0 0	0 0	0	0 0 0 1	0	0 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	0 1 1 1	0 1 1	0 0 1	0 0 0 1	0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	
0 0 0	0 0 0 0	0 0 0	0 0 1	0 1 1	1 1 1	1 1 1	1 1 1 1	1 1 1	1 1 1	1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	0 1 1 1 0	0 0 1 0	0 0 0 0	0 0 0 0 0	
0 0	0 0 0	0 0 0	0 0 0	0 0 0	0	0 0	1 :	1 1 1 0	1 1 1	1 1 1	1 1 1	1 1 1	1 1 1	1 1 1	1 1 1	1 1 0	1 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	
0 0 0	0 0 0 0	0 0 0	0 0 1	0 0 0	0 0 0	0 0 0	0 0 0 1	0	0 0 0	0 0 0	1 0 0	1 1 1 1	1 0 0	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	
0 0 0	0 0 0 0	0	1	1	1 1 1	1	1 :	1	1	1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1	0 0 0 0	0 0 0 0	
0 0 0	0	0 0 0	1 1 1	1 1 1	1 1 1	1 1 1	1 1 1 1 1 1	1 1 1	1 1 1	1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 0	1 1 0 0	0 0 0 0	0 0 0 0 0	
0 0	0 0 0	0 0 0	0 0 0	0 0	0 0	0 0	1 :	1 1 1 0	1 1 1	1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 1	1 1 1 0	1 1 0	1 0 0	1 0 0 0	0 0 0	0 0 0	0 0	0 0 0	0 0 0	
0 0 0		0	0	0	0	0	0	0		0	1 0 0	1	1 0 0	0	0 0 0	0 0 0	0 0 0	0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0 0	
0 0 0 0	0 0 0 0	0	0	0	0	0	0	0	0	0	0 0 0		0 0 0	0 0 0 0	0 0 0		0 0 0 0	0 0 0 0	0 0 0	0 0 0	8 0 0 0	0 0 0	0 0 0	
0 0 0	0 0 0 0	0 0 0	0 0	0 0 0	0 0	0 0 0	0	0 0 1	0 1 1 2	1 1 2	1 2 2 3 3	2 2 3 3 4	1 2 2 3 3	1 2 2 3	0 1 1 2 2	0 0 1 1 2	0 0 0 1 1	0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
0 0	0 0 0	0 0 0	0 0 1	0 1 1	1 1 2 2	2 2 3	2 :	2 3 4	3 4 4	3 4 4 5	4 5 5	4 5 5 6	4 5 5	3 4 4 5	3 4 4	2 3 4	2 3 3	1 2 2 3	1 2 2	0 1 1 2	0 0 1 0	0 0 0	0 0 0	
0 0 0	0 0 0	0 0 0	0 0	0 0	0 0	0 0	2 : 1 : 0 :	3 2 1 0	4 3 2 1	5 4 3 2	6 5 4 3	6 7 6 5 4	6 7 6 5	5 6 7 6	5 6 6 7	4 5 6 0	4 5 0	3 4 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0 0	
0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0 1	0	0 0 0	0	2 1 0 1	3 2 1 1	4 3 0 0	5 0 0 1	0 0 0 1 2	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 1	0 0 0 0	0 0 0	0 0 0	0 0 0 0	
0 0 0	0 0 0 0	0 0	1 1 1	2	3 3 3	4	3 4 5	3 4 5 6	3 4 5 6	2 3 4 5 6	2 3 4 5 6	2 3 4 5 6	2 3 4 5 6	3 4 5 6	3 4 5 6	2 3 4 5 6	2 3 4 5 5	2 3 4 4	2 3 3 3 3	2 2 2 2	1 1 1 1	0 0 0	0 0 0 0	
0 0 0	0 0 0 0	0	1 0 0	2 2 1	3 3 2	4 3 2	5 4 .	6 6 5	7 7 6 5	7 8 8 7 6	7 8 9 8 7	7 8 9 9	7 8 9 9	7 8 8 8	7 7 7 7 7	6 6 6	5 5 5 5	4 4 4	3 3 3 3	2 2 2 2	1 1 0 0	0 0 0 0	0 0 0 0 0	
0 0 0	0 0 0	0	0	0	0	0	0	2 1 0	3 2 1	5 4 3 2 1	6 5 4 3 2	7 6 5 4 3	8 7 6 5 4	8 7 6 5	7 7 7 7 0	6 6 0	5 0 0	4 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	
0 0	0 0 0	0	0	0	0	0	0	0	0	0 0 0	1 0 0	2 1 0	3 0 0	0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0 0	0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
	·									۰	٠	U	U	0	٠	۰			·	-	٠	U		

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	c
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	1	1	2	1	1	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	1	1	2	2	2	1	1	0	0	0	0	0	0	0	c
0	0	0	0	0	0	0	0	1	1	2	2	3	2	2	1	1	0	0	0	0	0	0	c
0	0	0	0	0	0	0	1	1	2	2	3	3	3	2	2	1	1	0	0	0	0	0	c
0	0	0	0	0	0	1	1	2	2	3	3	4	3	3	2	2	1	1	0	0	0	0	c
0	0	0	0	0	1	1	2	2	3	3	4	4	4	3	3	2	2	1	1	0	0	0	c
0	0	0	0	1	1	2	2	3	3	4	4	5	4	4	3	3	2	2	1	1	0	0	c
0	0	0	1	1	2	2	3	3	4	4	5	5	5	4	4	3	3	2	2	1	1	0	c
0	0	0	0	1	1	2	2	3	3	4	4	5	4	4	3	3	2	2	1	1	0	0	c
0	0	0	0	0	1	1	2	2	3	3	4	4	4	3	3	2	2	1	1	0	0	0	C
0	0	0	0	0	0	1	1	2	2	3	3	4	3	3	2	2	1	1	0	0	0	0	C
0	0	0	0	0	0	0	1	1	2	2	3	3	3	2	2	1	1	0	0	0	0	0	C
0	0	0	0	0	0	0	0	1	1	2	2	3	2	2	1	1	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	1	1	2	2	2	1	1	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	1	1	2	1	1	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	C
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	C
0	0	0	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	0	C
0	0	0	1	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	1	0	C
0	0	0	1	2	3	4	4	4	4	4	4	4	4	4	4	4	4	4	3	2	1	0	C
0	0	0	1	2	3	4	5	5	5	5	5	5	5	5	5	5	5	4	3	2	1	0	C
0	0	0	1	2	3	4	4	5	5	6	6	6	6	6	5	5	4	4	3	2	1	0	C
0	0	0	1	2	3	3	4	4	5	5	6	6	6	5	5	4	4	3	3	2	1	0	C
0	0	0	1	2	2	3	3	4	4	5	5	6	5	5	4	4	3	3	2	2	1	0	C
0	0	0	1	1	2	2	3	3	4	4	5	5	5	4	4	3	3	2	2	1	1	0	C
0	0	0	0	1	1	2	2	3	3	4	4	5	4	4	3	3	2	2	1	1	0	0	C
0	0	0	0	0	1	1	2	2	3	3	4	4	4	3	3	2	2	1	1	0	0		C
0	0	0	0	0	0	1	1	2	2	3	3	4	3	3	2	2	1	1	0	0	0	0	C
0	0	0	0	0	0	0	1	1	2	2	3	3	3	2	2	1	1	0	0	0	0	0	C
0	0	0	0	0	0	0	0	1	1	2	2	3	2	2	1	1	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	1	1	2	2	2	1	1	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	1	1	2	1	1	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	c

2nd pass distance transform with choice = 8

| Color | Colo

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(
0	0	0	0	0		0	0	0	0	0		5	0	0	0	0	0	0	0	0	0	0	(
0	0	0	1	0	2	0	3	0	4	0	5	5	5	0	4	0	3	0	2	0	1		(
0						0										0					0		(
0	0		0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0		(
0	0		0	0		0	0			0		4		0			0		0	0	0		(
0	0	0		0		0	0			0		0		0			0		0		0		(
0		0				0		0								0					0		(
0		0				0															0	0	
	0					0		0		0			0				0		0		0		(
0	0	0	0	0	0	0	0	0	0	0		0			0	0	0	0	0	0	0		(
0	0	0	0	0		0		0		0		1	0		0		0		0	0	0		(
0	0	0				0				0		1		0			0		0		0		(
0	0	0				0					0						0		0		0		0
	0		0	0		0		0		0		0			0		0		0	0	0		0
0	0	0		0		0	0			0		0		0			0		0	0	0		0
		0			0		5			0			0				5		0		0		
0						0										5					0	0	
	0					0		0		0		6				0			0		0	0	
0	0	0	0	0		0	0	0				6		0		0	0	0	0	0	0		Ċ
0	0	0	0	0		0		0		0			0		0		0		0	0	0		
0	0	0				0				0		5		0			0		0		0		Ċ
0		0				0										0		ō			0		ì
	0					0		0								0			0		0		(
0	0		0	0		0		0				0			ō		0	0	0	ō	0		(
0		0	0	0		0	0			0		3		0			0		0	ō	0		(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	o	0	0	0	0	o	0		(
0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	o	0	0	0	0	o	0		(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0		(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		(
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	(

Below is the loaded skeleton with choice = 8

2	nd	n	985		×	na:	าลา	io		wi:	t.h	cì	201	io		- :	R						
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	ō	0	0	0
0	0	0	0	0	ō	0	0	0	0	0	1	1	1	0	0	0	0	0	0	ō	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	2	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	2	2	2	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	2	2	3	2	2	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	2	2	3	3	3	2	2	1	1	0	0	0	0	0	0
0	0	0	0	0	0	1	1	2	2	3	3	4	3	3	2	2	1	1	0	0	0	0	0
0	0	0	0	0	1	1	2	2	3	3	4	4	4	3	3	2	2	1	1	0	0	0	0
0	0	0	0	1	1	2	2	3	3	4	4	5	4	4	3	3	2	2	1	1	0	0	0
0	0	0	1	1	2	2	3	3	4	4	5	5	5	4	4	3	3	2	2	1	1	0	0
0	0	0	0	1	1	2	2	3	3	4	4	5	4	4	3	3	2	2	1	1	0	0	0
0	0	0	0	0	1	1	2	2	3	3	4	4	4	3	3	2	2	1	1	0	0	0	0
0	0	0	0	0	0	1	1	2	2	3	3	4	3	3	2	2	1	1	0	0	0	0	0
0	0	0	0	0	0	0	1	1	2	2	3	3	3	2	2	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	2	2	3	2	2	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	2	2	2	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	2	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	0	0
0	0	0	1	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	2	1	0	0
0	0	0	1	2	3	4	4	4	4	4	4	4	4	4	4	4	4	4	3	2	1	0	0
0	0	0	1	2	3	4	5	5	5	5	5	5	5	5	5	5	5	4	3	2	1	0	0
0	0	0	1	2	3	4	4	5	5	6	6	6	6	6	5	5	4	4	3	2	1	0	0
0	0	0	1	2	3	3	4	4	5	5	6	6	6	5	5	4	4	3	3	2	1	0	0
0	0	0	1	2	2	3	3	4	4	5	5	6	5	5	4	4	3	3	2	2	1	0	0
0	0	0	1	1	2	2	3	3	4	4	5	5	5	4	4	3	3	2	2	1	1	0	0
0	0	0	0	1	1	2	2	3	3	4	4	5	4	4	3	3	2	2	1	1	0	0	0
0	0	0	0	0	1	1	2	2	3	3	4	4	4	3	3	2	2	1	1	0	0	0	0
0	0	0	0	0	0	1	1	2	2	3	3	4	3	3	2	2	1	1	0	0	0	0	0
0	0	0	0	0	0	0	1	1	2	2	3	3	3	2	2	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	2	2	3	2	2	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	2	2	2	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	2	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

deCompressedFile for img1 using 8-distance logFile for img1 using 8-distance

PrettyPrint for img2 using	8-distance
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	
	0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	$\begin{smallmatrix}0&0&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0$
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	$1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\$
	$f1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \$
	1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 0
	$1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0$
	1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0
0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 1 1 1 1
	$0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0$
	$0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\$
0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0
	$0\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\$
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 1 1 1 1 0 0 0
	$1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 0\ 0\ 0$
	f1111111111111111000000100000000000000
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
	$1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\$
	$f1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ $
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	$0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\$
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	$0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \$
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	$\begin{smallmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 &$
000000000000000000000000000000000000000	1 1 2 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
	$1\ 2\ 2\ 3\ 2\ 2\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\$
	$2\ 3\ 3\ 4\ 3\ 3\ 2\ 2\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\$
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	3 4 4 5 4 4 3 3 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	$4\ 4\ 5\ 5\ 5\ 4\ 4\ 3\ 3\ 2\ 2\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\$
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	$5\ 5\ 6\ 6\ 6\ 5\ 5\ 4\ 4\ 3\ 3\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 0\ 0$
0 0 0 0 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 0 0 0 0	3 4 5 6 7 6 6 5 5 0 0 0 0 0 0 0 0 1 2 1 1 0 0 0 0 0 0 0 1 1 2 1 0 0 0
0 0 0 0 1 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 1 0 0 0 0 0 0 1 2 3 4 4 4 4 4 4 4 4 4 4 4 3 2 1 0 0 0 0 0 0 0 0 0	$\begin{smallmatrix}2&3&4&5&6&7&6&6&0&0&0&0&0&0&0&0&1&2&2&1&1&0&0&0&0&0&1&1&2&2&1&0&0&0\\1&2&3&4&5&6&7&0&0&0&0&0&0&0&0&0&1&2&2&2&1&1&0&0&0&0&1&1&2&2&2&1&0&0&0\\1&2&3&4&5&6&7&0&0&0&0&0&0&0&0&0&1&2&2&2&1&1&0&0&0&0&1&1&2&2&2&1&0&0&0\\1&2&3&4&5&6&7&0&0&0&0&0&0&0&0&0&1&2&2&2&1&1&0&0&0&0&1&1&2&2&2&1&0&0&0\\1&2&3&4&5&6&7&0&0&0&0&0&0&0&0&0&1&2&2&2&1&1&0&0&0&0&0$
	$0\ 1\ 2\ 3\ 4\ 5\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 3\ 2\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 2\ 2\ 3\ 2\ 1\ 0\ 0\ 0$
0 0 0 0 1 2 3 4 5 6 7 7 7 7 7 6 5 4 3 2 1 0 0 0 0 0 0 0	0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 2 3 3 3 2 2 1 1 1 2 2 3 3 3 2 1 0 0 0
	0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	$0\ 1\ 1\ 2\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 2\ 3\ 4\ 4\ 4\ 4\ 4\ 0\ 1\ 2\ 3\ 4\ 4\ 3\ 2\ 1\ 0\ 0\ 0$
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	1 2 2 3 2 2 1 1 0 0 0 0 0 0 0 0 0 1 2 3 4 5 6 0 0 0 0 1 2 3 3 2 1 0 0 0
000000000000000000000000000000000000000	2
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	$3\; 3\; 4\; 4\; 4\; 3\; 3\; 2\; 2\; 1\; 1\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 1\; 2\; 3\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 1\; 2\; 1\; 0\; 0\; 0$
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	4 4 5 5 5 4 4 3 3 2 2 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
	4 5 5 6 5 5 4 4 3 3 2 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	4 5 6 7 6 6 5 5 4 4 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	2 3 4 5 6 7 6 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	1 2 3 4 5 6 7 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$ \begin{array}{cccccccccccccccccccccccccccccccccccc$	0 1 2 3 4 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	0 0 1 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix}0&0&1&2&3&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix}0&0&1&2&3&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	0 0 1 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	0 0 1 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$\begin{smallmatrix} 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$

| Section High Plane | Section

0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 (0 0	0	0	0 (0	0	0	0 0	0	0	0	0 (0 0	0	0	0	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 (0
0	0 0	0	0	0 0	0 0	0	0	0 0	0	0	0	0	0 0	0	0	0 (0	0	0	0 0	0	0	0	0 (0 0	0	0	0	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 (0
0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0	0 0	0	0	0 (0	0	0	0 0	0	0	0	0 (0 0	0	0	0	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 (0
0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0	0 0	0	0	0 (0	0	0	0 0	0	0	0	0 (0 0	0	0	0	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 (0
0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0	0 0	0	0	0 (0	0	0	0 0	0	0	0	0 :	1 0	0	0	0 1	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 (0
0	0 0	0	0	0 0	0 0	0	0	0 0	0	0	0	0 1	0 0	0	0	0 (0	0	0	0 0	0	0	0	0 (0 0	0	0	0 1	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 (0
0	0 0	0	0	0 0	0 0	0	0	0 0	0 0	0	0	0 1	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 2	2 0	0	0	0 1	0 0	0	0	0 0	0	0	0 0	0	0 0	0	0	0 0	0	0	0	0 0	0	0	0	0 0	0	0 1	Ô
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
																																		0																	
U	0	U	U	0 (. 0	U	U	0 (, (. 0	U	0 1	0 (, 0	U	0 (, 0	0	U	U U	. 0	U	U	0 (0	U	U	0 1	0	. 0	U	0 0	, 0	U	U U	. 0	U	, 0	U	u C	, 0	U	U	0 (, 0	U	U	0 0	0	0 1	U

Below is the loaded skeleton with choice = 8

2nd pass expansion with choice = 8	
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	100000000000000000000000000000000000000
	110000000000000000000000000000000000000
	2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	3 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	3 3 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	4 3 3 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	4 4 3 3 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	5 4 4 3 3 2 2 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
	4 4 3 3 2 2 1 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
	4 3 3 2 2 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
	$\begin{smallmatrix} 3 & 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1$
	$\begin{smallmatrix} 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
	$\begin{smallmatrix}2&2&1&1&0&0&0&0&0&0&0&0&0&1&2&2&1&1&0&0&0&0$
	$\begin{smallmatrix}2&1&1&0&0&0&0&0&0&0&0&0&0&1&2&2&2&1&1&0&0&0&0$
	$1\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;2\;3\;2\;2\;1\;1\;0\;0\;0\;1\;1\;2\;2\;3\;2\;1\;0\;0\;0$
	$\begin{smallmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 &$
	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
	$\begin{smallmatrix}1&0&0&0&0&0&0&0&0&0&0&0&0&1&2&3&3&3&2&2&1&1&1&2&2&3&3&3&2&1&0&0&0\end{smallmatrix}$
	$1\;1\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;0\;1\;2\;3\;3\;2\;2\;1\;1\;0\;1\;1\;2\;2\;3\;3\;2\;1\;0\;0\;0$
	$\begin{smallmatrix}2&1&1&0&0&0&0&0&0&0&0&0&0&1&2&3&2&2&1&1&0&0&0&1&1&2&2&3&2&1&0&0&0\end{smallmatrix}$
	$\begin{smallmatrix}2&2&1&1&0&0&0&0&0&0&0&0&0&1&2&2&2&1&1&0&0&0&0$
	$\begin{smallmatrix} 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
	$\begin{smallmatrix} 3 & 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1$
	$4\; 3\; 3\; 2\; 2\; 1\; 1\; 0\; 0\; 0\; 0\; 0\; 0\; 1\; 1\; 1\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 1\; 1\; 1\; 0\; 0\; 0$
	$\begin{smallmatrix} 4 & 4 & 3 & 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0$
	$\begin{smallmatrix}5&4&4&3&3&2&2&1&1&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 4 & 4 & 3 & 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$4\; 3\; 3\; 2\; 2\; 1\; 1\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\; 0\;$
	$\begin{smallmatrix} 3 & 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 3 & 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 2 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$1 \; 1 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; $
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 &$
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
	$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
$\begin{smallmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 $	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

SkeletonFlle for img2 using 8-distance	
6401 311	
31 2 313	
314 215	
1921 1942	
1263	
284 30 5	
315 325	
344 363	
382 401	
1315 1461	
621 6461	
5 47 1	
6611 621	
3314 472	
61 2 313	
1.472 1.482	
662 8612	
312	
483 603	
196 1106	
116 126	
136 146	
156 483	
493	
593 603	
1311 1311	
494 1613	
3 53 2 3 54 2	
552 3573	
1594	
312 483	
49 3 55 3	
603 483	
660 3 '313	
. 472 . 482	
1602	
612 314	
472 612	
315 461	
471 611	
1 62 1 2 2 2 1	
242 2263	
284	
305 315	
325 344	
363 382	
2.401 2.461	
3314	
313 312	

deCompressedFile for img2 using 8-distance

logFile for img1 using 4-distance

Entering DistanceTransform() method Leaving DistanceTransform() method Entering compression() method. In compression() Below is skeleton A

Leaving compression() method Entering deCompression() method