[MS1] Intelligent Scissors

# Sample Tests

|  |  |  |  |
| --- | --- | --- | --- |
| Case # | Input Image | Image width and height | Output file |
| 1 | Case1/test1.png | 11x11 | Case1/output.txt |
| 2 | Case2/test2.png | 100x100 | Case2/output.txt |
| 3 | Case3/Case1.1.jpg | 78x59 | Case3/output.txt  Case3/ShortestPath.txt |

## Cases Output Description

You shall output these for each case in milestone 1:

1. Output file contain the construction of graph
   1. Mention every index node which represents every pixel in the image(Hint: the index of pixel after converting the image from 2d to 1d)
   2. With the edges from this index node to the surrounding neighbors and the weight for each edge(**Hint : using 4 connectivity**)

2. Path file contains the list of pixels that represent the shortest path between the source node and the destination node (**Hint: Saving the node index besides the pixel X’s and Y’s position**)

**Check the output of each case below …**

* **Case 1:**
* **Case-1/test1.png.**
* **Output.txt** shall contain:
  + **The graph that is constructed from the image**
  + **Every node index which represents one pixel in the image followed by the edges between it and the neighbors with the weights**
  + **As following**

The index node 118

Edges

edge from 118 To 119 With Weights 1E+16 (Hint when the result is infinitely)

edge from 118 To 107 With Weights 1E+16

edge from 118 To 117 With Weights 1E+16

## Case 2:

* **Case-1/test2.png.**
* **Output2.txt** shall contain:
  + **The graph that is constructed from the image**
  + **Every node index which represents one pixel in the image followed by the edges between it and the neighbors with the weights**
  + **As following**

## The index node1

## Edges

## edge from 1 To 2 With Weights 1E+16

## edge from 1 To 101 With Weights 1E+16

## edge from 1 To 0 With Weights 1E+16

## Case 3:

* **Case-1/cas41.1.jpg.**
* **Output3.txt** shall contain:
  + **The graph that is constructed from the image**
  + **Every node index which represents one pixel in the image followed by the edges between it and the neighbors with the weights**
  + **As following**

## The index node2

## Edges

## edge from 2 To 3 With Weights 0.999999999999972

## edge from 2 To 80 With Weights 0.333333333333333

## edge from 2 To 1 With Weights 0.499999999999993

* **the file shortestPath .txt shall contains the list of pixels that represent the shortest path between the source node and the destination node as following**

ode 102 at position x 24 at position y 1

Node 103 at position x 25 at position y 1

Node 104 at position x 26 at position y 1

Node 26 at position x 26 at position y 0

Node 27 at position x 27 at position y 0

Node 28 at position x 28 at position y 0

Node 108 at position x 30 at position y 1

Node 32 at position x 32 at position y 0

Node 110 at position x 32 at position y 1

Node 32 at position x 32 at position y 0

Node 33 at position x 33 at position y 0

Node 113 at position x 35 at position y 1

Node 114 at position x 36 at position y 1

Node 115 at position x 37 at position y 1

Node 39 at position x 39 at position y 0