

INTRO TO ALGORITHMS ASSIGNMENT 3 GROUP TASK

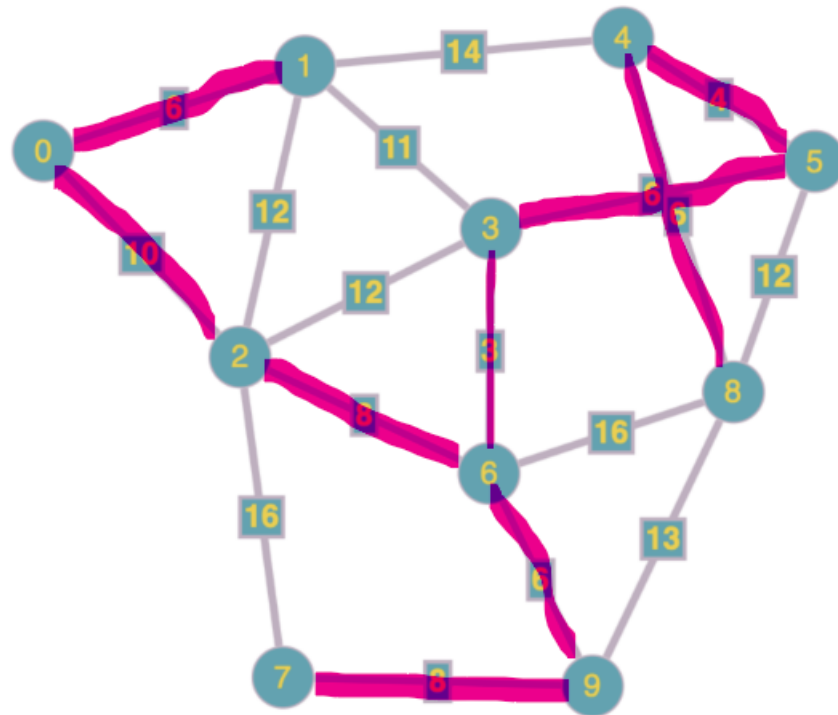
ALEX KUNZLER & PEYTON GRATTINO

CONTENTS

1. Task 2	2
2. Efficiency Heap Algorithm	2

1. TASK 2

	0	1	2	3	4	5	6	7	8	9
0	∞	6	10	∞	∞	∞	∞	∞	∞	∞
1	6	∞	12	11	14	∞	∞	∞	∞	∞
2	10	12	∞	12	∞	∞	8	16	∞	∞
3	∞	11	12	∞	∞	6	3	∞	∞	∞
4	∞	14	∞	∞	∞	4	∞	∞	6	∞
5	∞	∞	∞	6	4	∞	∞	∞	12	∞
6	∞	∞	8	3	∞	∞	∞	∞	16	6
7	∞	∞	16	∞	∞	∞	∞	∞	∞	8
8	∞	∞	∞	∞	6	12	16	∞	∞	13
9	∞	∞	∞	∞	∞	∞	6	8	13	∞



2. EFFICIENCY HEAP ALGORITHM

```
start node x //the head of the tree
Go to the left node //the left side should always be less than x
REPEAT UNTIL CAN'T GO FURTHER
Go down smallest valued path
if smallest node is last node
    delete
else
    swap this node with the rightmost leaf on the left side
    delete
```

Use a sortheap command to check, and resort to make sure that it is still a heap.