Assignment 2

Question 1:

* Breadth First Search
- Order of Visited Nodes:
  A, B, C, D, E, F

* Depth First Search
- Order of Visited Nodes:
  A, B, D, C, E, f

* Iterative Deepening search
- Order of visited Nodes:
  when limit = 0 ⇒ A

  when limit = 1 ⇒ A, B, C

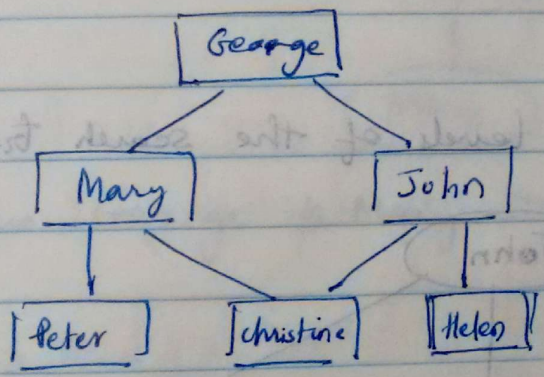
  when limit = 2 ⇒ A, B, D, C, E, F

* Uniform cost search
- Order of visited Nodes:
  A, C, E, B, D, F

Question 2

(i) Uniform cost search, and Iterative deepening search would guarantee finding the correct number of degrees of seperation between any 2 people in the graph.

For Breadth First Search,



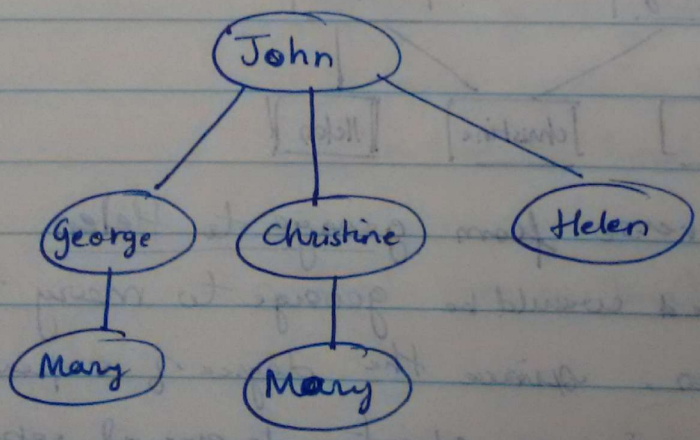
To traverse from george to Helen path followed would be george to mary to christine to john to helen, hence the degree of seperation = 4.
but there is a shorter degree of separation from george to john to Helen, degree of seperation = 2
∴ BFS fail to give minimum degree of seperation

Whereas Uniform cost algorithm tries to find the minimum cost path, so it will follow george to john to helen with minimum degree of seperation = 2

For DFS, it will follow george - mary - christine - John - Helen path with degree of seperation = 4, which is not the optimal path. hence DFS fail to provide minimum degree of seperation.

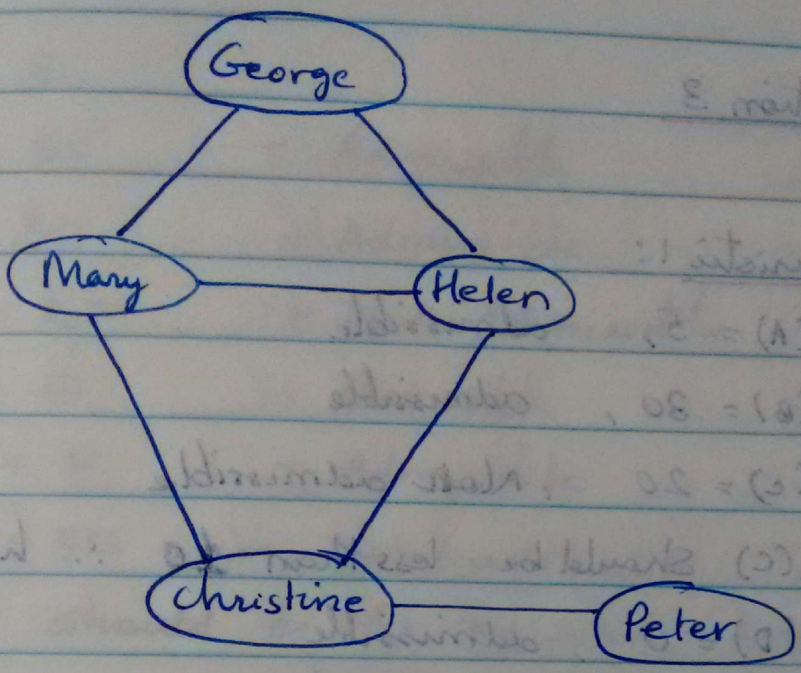
Iterative deepening search will follow the path from george - John - helen with min degree of seperation = 2.

(ii) First three levels of the search tree,



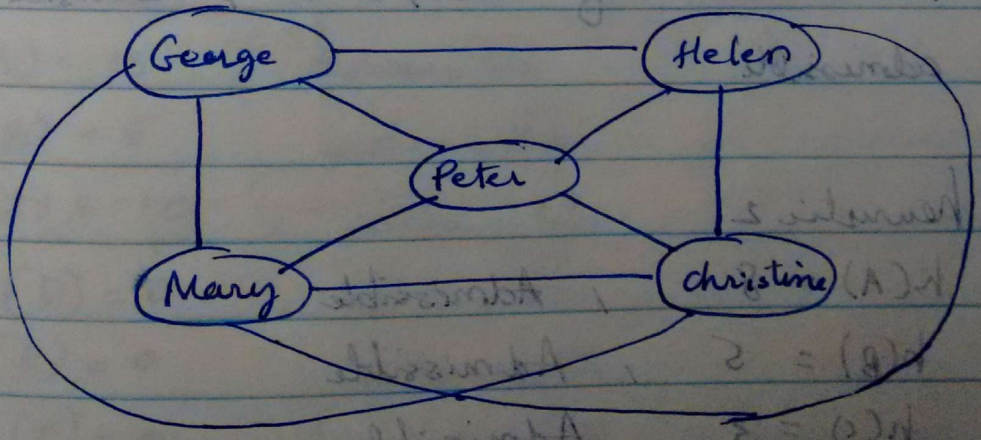
There is no one to one correspondence between the nodes in the search tree and vertices of SVG. since in the search tree the vertice "Peter" it is not reached.

(iii)



Here George & Peter have 4 degree of seperation via George — Mary — Helen - Christine — Peter

(iv)



v) As we need to find the degrees of seperation between 1 million people in SNG using BFS, the BFS search tree algorithm has the space complexity $O(b^d)$ as we will have to traverse the entire tree.

Given in the problem, storing 1 node requires 1 kb of memory, and the memory should not exceed $\underline{1GB = 10^6 kb}$, 1 million people equals 10^6 people.

If the BFS is used with the branching factor $b = \underline{10}$, & so the depth(d) equals $\underline{6}$, using the space complexity

$O(b^d) = 10^6$ kb will be required.

So we will be able to traverse the tree using BFS algorithm without exceeding 1 GB.

Question 3

* **Heuristic 1:**

$h(A) = 5$, admissible

$h(B) = 30$, admissible

$h(C) = 20$, Not admissible

$h(C)$ Should be less than 10 $\therefore h(C) \leq 10$

$h(D) = 0$, admissible

$h(E) = 10$, admissible

$h(F) = 0$, admissible

Heuristic 1 is not admissible, but if we adjust the value of $h(C) = 10$, heuristic 1 becomes admissible.

* **Heuristic 2**

$h(A) = 8$, Admissible

$h(B) = 5$, Admissible

$h(C) = 3$, Admissible

$h(D) = 5$, Admissible

$h(E) = 5$, Admissible

$h(F) = 0$, Admissible

$\therefore$ heuristic 2 is admissible

* heuristic 3 :

h(A) = 35 - Admissible

h(B) = 30 - Admissible

h(C) = 20 - Non-Admissible, $h(c) \leq 10$.

h(D) = 0 - Admissible

h(E) = 0 - Admissible

h(F) = 50 - Non-admissible

h(F) should be less than = 40, $h(F) \leq 40$ to become admissible.

h(B) should be less than equal to 30,

h(B) = 10, becomes admissible.

* heuristic 4 :

h(A) = 15, admissible

h(B) = 5, admissible

h(C) = 10, admissible

h(D) = 0, admissible

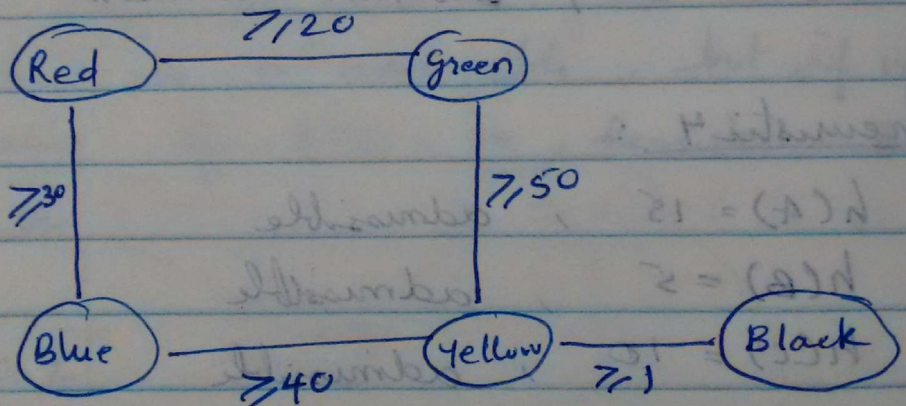
h(E) = 0, admissible

h(F) = 0, admissible

* Heuristic 5 :

$h(A) = 0$, Admissible
$h(B) = 0$, Admissible
$h(C) = 0$, Admissible
$h(D) = 0$, Admissible
$h(E) = 0$, Admissible
$h(F) = 0$, Admissible

Question 4 :



Assuming ~~dis~~ heuristic distance from yellow to Black = 1

* $h(red)$; to reach from red to black, there 2 paths, red - green - yellow - black ≥ 71
red - blue - yellow - black ≥ 71
$\therefore h(red) \geq 71$.

* For h(green),
from green to black again there are 2 paths
green - red - blue - yellow - black ≥ 91
green - yellow - black ≥ 51

∴ h(green) = 51

* h(blue)
To traverse from blue to black there are 2 paths
Blue to yellow to Black ~~or~~ which is ≥ 41,
blue to red to green to yellow to black ≥ 101

∴ h(blue) = 41 .

* h(yellow)
The lowest possible distance from yellow to black would be 1.
∴ h(yellow) ≥ 1

* h(black) = 0.
The cost it takes to reach black from black is 0.

Question 5

(a) None of the algorithms like breadth-first search, depth-first search, iterative deepening search, uniform cost search, A* and IDA* can gurantee that the goal state can be reached within 50KB memory usage because it stays for some initial states, the shortest solution is longer than 1000 moves. And each move will consist of a new state and hence require 1KB of memory. So these algorithm will need atleast 1000KB of memory when reaching the goal state.

(b) Iterative deepening search and IDA* will can gurantee that it will not need more than 1200KB of memory to store search nodes.
Because of Incase of breadth first search the space complexity grows exponentially, and like it tried to visit all the nodes in the search tree until the goal is reached.
The DFS algorithm, it may be possible that it might get into infinite loop, hence it cannot gurantee.

The space complexity of uniform cost search algorithm is $O(b^{[C^*/\epsilon]})$ which is exponential. and hence will not gurantee memory usage within 1200kB.
The Space complexity of A* is $O(b^d)$ which is again exponential therefore it does not gurantee that it will not need more than 1200kB of memory.
whereas the IDS algorithm and IDA* search algorithms have linear space complexities so, it will gurantee that it will not take 1200kB of memory.