Design & Analysis of Algorithm
Final Examination (Spring 2022)

Reg. No: _____ Time: 150 mins

Q1: (10 Marks)

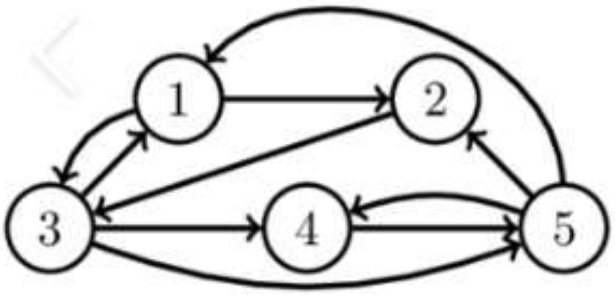
Write a function that sorts a character array of size **n** in decreasing order. Each character in the array is either an 'R', 'G' or 'B'. These three characters will be present in the array.

The list can only be traversed once. It is not allowed to use any extra array/List/Stack etc. . The time complexity of the algorithm shall be **O(n)**.

Example:
Input: RBBGBGRRGBRGB
Output: RRRRGGGGBBBBB

Q2: (8+7 Marks)

```
BFS(G, s)
 1  for each vertex u ∈ G.V − {s}
 2      u.color = WHITE
 3      u.d = ∞
 4      u.π = NIL
 5  s.color = GRAY
 6  s.d = 0
 7  s.π = NIL
 8  Q = ∅
 9  ENQUEUE(Q, s)
10  while Q ≠ ∅
11      u = DEQUEUE(Q)
12      for each v ∈ G.Adj[u]
13          if v.color == WHITE
14              v.color = GRAY
15              v.d = u.d + 1
16              v.π = u
17              ENQUEUE(Q, v)
18      u.color = BLACK
```

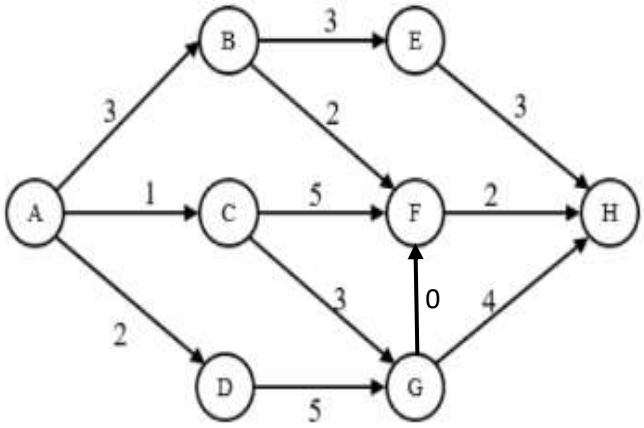


Apply BFS algorithm on the given graph and answer the following questions. The source node is '**5**'. (**Note**: The adjacent nodes of a vertex are to be traversed in increasing order.)

a) Write the type of each edge of the above graph.
b) Write the names of the nodes/vertices that are in the queue '**Q**' when L17 is executed for the 4th time and also write the values of their '**distance'**, '**predecessor'** and '**color'** attributes.

Q3: (9+7 Marks)

```
DIJKSTRA (G, w, s)
1.  INITIALIZE-SINGLE-SOURCE(G, s)
2.  X = Ø
3.  H = G.V
4.  While H ≠ Ø
5.      u = EXTRACT-MIN(H)
6.      X = X U {u}
7.      for each vertex v ∈ G. Adj[u]
8.          if v. d > u. d + w(u, v)
9.              v. d = u. d + w(u, v)
10.             v. π = u
```



Apply Dijkstra's Shortest Path Algorithm on the given graph and answer the following questions. The source node is '**A**'. (**Note**: The adjacent nodes of a vertex are to be traversed in increasing order.)

a) Write are the values of '**u**', '**u.d**', '**u.π**', '**v**', '**v.d**', and '**v.π**' , when the attribute values of '**F**' are changed for the third time.
b) What are the '**distance'** and '**predecessor'** values of nodes that are not in the heap, **H**, when L10 is executed for the 6th time.

Q4: (5+5 Marks)

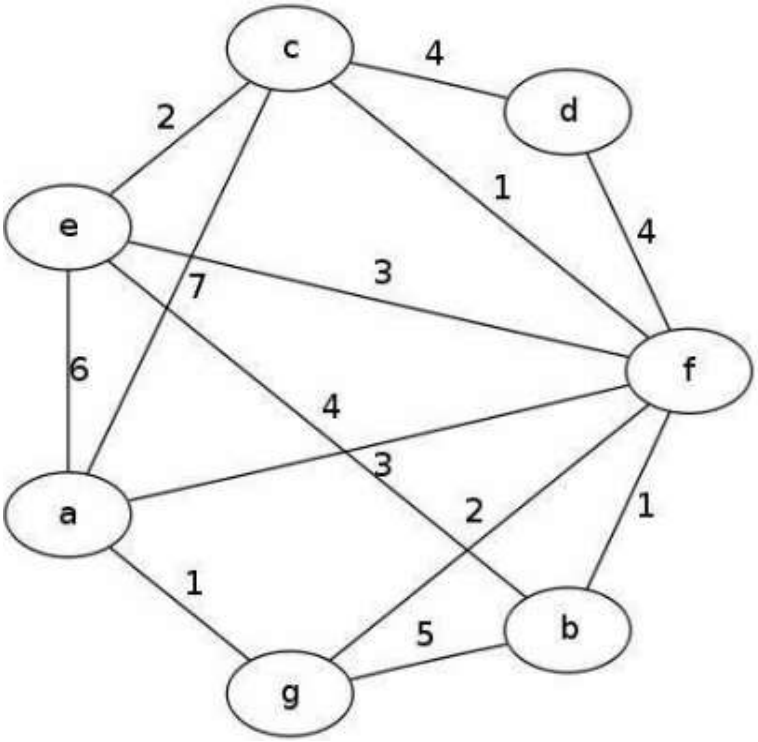
```
INSERTION-SORT(A)
1  for j = 2 to A.length
2      key = A[j]
3      // Insert A[j] into the sorted sequence A[1 .. j − 1].
4      i = j − 1
5      while i > 0 and A[i] > key
6          A[i + 1] = A[i]
7          i = i − 1
8      A[i + 1] = key
```

Solve the following for A = {5, 4, 2, 6, 1, 3}. (**Note**: **Indexing** starts from **1**.)

a) Write the value of '**key'** and contents of '**A**' when **L8** is reached for the 2nd time (but not executed).
b) Write the values of '**key', 'A[i]'** and contents of '**A**' when **L6** is executed for the 6th time.

Q5: (2+3+5+5 Marks)

```
MST-PRIM(G, w, r)
 1.  for each u ∈ G.V
 2.    u. key = ∞
 3.    u. π = NIL
 4.  r. key = 0
 5.  Q = G.V
 6.  while Q ≠ Ø
 7.    u = EXTRACT-MIN(Q)
 8.    for each v ∈ G. Adj[u]
 9.        if v ∈ Q and w(u, v) < v. key
10.            v. π  = u
11.            v. key = w(u, v)
```



Apply Prim's MST Algorithm on the given graph and answer the following questions. The source node is '**a**'.

(**Note**:

- The adjacent nodes of a vertex are to be traversed in increasing order.
- The weights of edges {b, e} and {a, f} are 4 and 3 respectively.
- When more than one node of equal 'key' values are candidates to be extracted from the heap, choose the node with lowest name.

)

a) What is the weight of the MST?
b) Draw the partial MST when node '**e**' is extracted from the heap, Q, in Line7.
c) Write edges that are rejected by the algorithm when node '**e**' is extracted from the heap, Q, in Line7.
d) Write the '**key**' and '**π**' attributes of nodes that are not in the heap, Q, when node '**b**' is extracted from the heap in Line7.