National University of Computer & Emerging Sciences, Karachi
Fall-2021 Department of Computer Science
Final Exam
29 December 2021, 09:00 AM – 12:00 AM
Part (B)

Course Code: CS2009	**Course Name:** Design and Analysis of Algorithm
Instructor Name / Names: Dr. Muhammad Atif Tahir, Dr. Fahad Sherwani, Dr. Farrukh Saleem, Waheed Ahmed, Waqas Sheikh, Sohail Afzal	
Student Roll No:	**Section:**

Instructions:

- Return the question paper.
- Read each question completely before answering it. There are **10 questions** on **4 pages**.
- In case of any ambiguity, you may make assumption. But your assumption should not contradict any statement in the question paper.
-

Time: 150 minutes **Max Marks: 40**

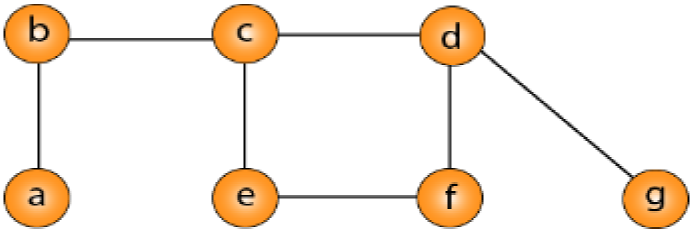
Question # 4 **[6 marks]**

a) What is meant by P and NP Problems?

b) Let X be a problem that belongs to the class NP. Then explain why the following are incorrect or correct statements?
   - (Explain why it is incorrect) There is no polynomial time algorithm for X
   - (Explain why it is incorrect) If X can be solved deterministically in polynomial time then P = NP
   - (Explain why it is correct) If X is NP-hard, then it is NP-complete
   - (Explain why it is incorrect) X may be undecidable

c) Does P ! = NP mean that no problem exists which can be solved and checked in polynomial time?

d) Using 2-approximation greedy method studied during lectures, find the size of the vertex cover of the following graph:



Question # 5

[4 marks]

Prove that Travelling Salesman approximation algorithm is a 2-approximation algorithm**.** Give arguments and example as well.

Question # 6 **[4 marks]**

Given a directed graph G = (V, E) where each edge (u,v) ϵ E has an associated value r(u,v), which is a real number in the range $0 \leq r(u,v) \leq 1$ that represents the reliability of a communication channel from vertex u to vertex v. We interpret r(u,v) as the probability that the channel from u to v will not fail, and we assume that these probabilities are independent. Give an efficient algorithm to find the most reliable path between two given vertices.

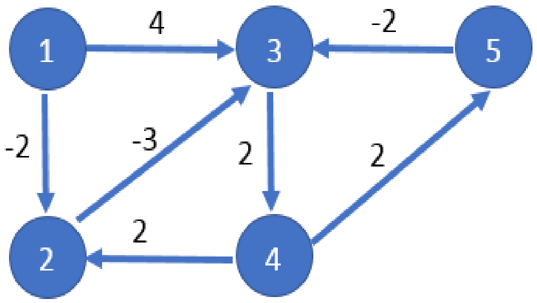
(Note: Reliability means higher value)

Question # 7 **[3 + 1 + 1= 5 marks]**

(a) Use Floyd Warshall algorithm to find the shortest path from every vertex to every other vertex for the graph given below. The result must contain two matrices/tables, one matrix (D) shows the shortest cost from each vertex to all other vertices, and the second matrix (Π) should show the previous vertex use to reach the destination vertex.

(b) Mark the steps on the resultant Π matrix/table to show, how it will be used, to trace the route from vertex 5 to vertex 2.

(c) From the resultant matrices, identify all those pairs of vertex for which there is no path.



Question # 8 **[2+3 = 5 marks]**

Let array A= $A[1], A[2], \ldots, A[n]$ stores the number of cars produced by a company in the past n years,. The company wants to find if there is a period of consecutive years (from i to j), such that the total number of cars produced in this period is exactly equal to M.

if there exist i and j, $1 \le i \le j \le n$, such that $\sum_{k=i}^{j} A[k] = M$.

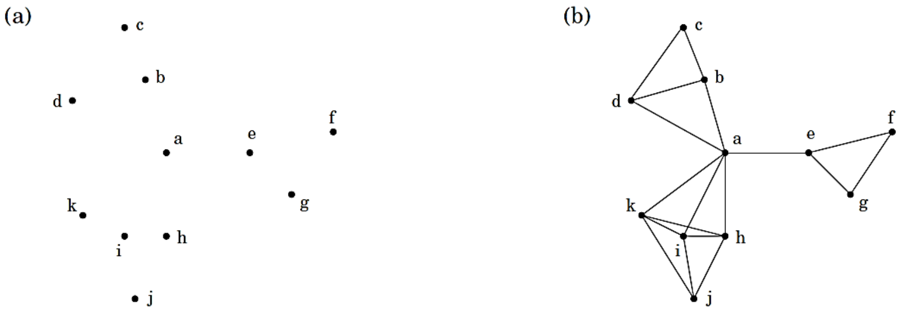
Example : Input : A [2, 4, 6, 3, 8, 4, 1, 10], M= 15

a) Design brute force algorithm for this problem
b) Design O(n) time algorithm to solve this problem.

Question # 9 **[5 marks]**

The dots in below *Figure(a)* represent a collection of towns, and the edges between the towns in *Figure(b)* shows that the two towns u and v with the edge (u,v) are at-most 30 miles apart. The town committee is deciding where to open the schools.
There are only two constraints:
a) Each school should be in a town.
b) No student should have to travel more than 30 miles to reach one of the school.

With the above given constraints, design approximation algorithm to determine the minimum number of schools needed.

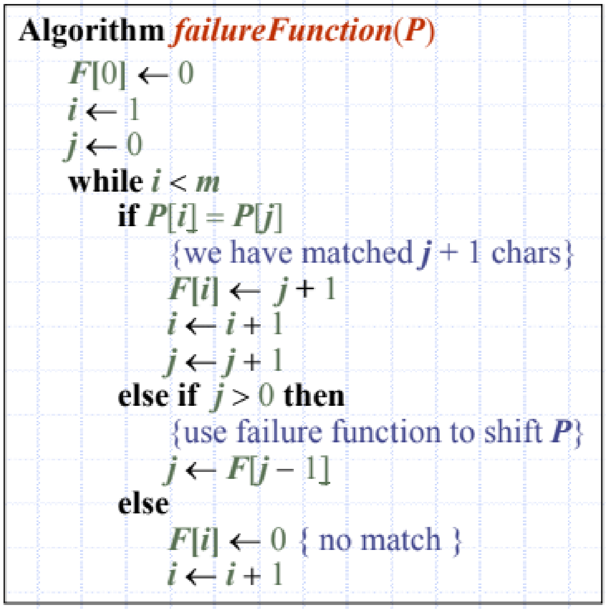


Question # 10 **[3 marks]**

Construct failure function table (that we build up in KMP string matching algorithm) for given pattern :

T:	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

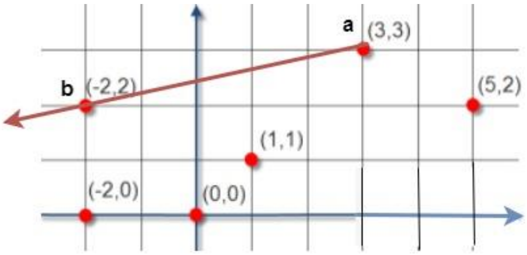
P:	a	b	a	b	a	c	a



```
Algorithm failureFunction(P)
    $F[0] \leftarrow 0$
    $i \leftarrow 1$
    $j \leftarrow 0$
    while $i < m$
        if $P[i] = P[j]$
            {we have matched $j + 1$ chars}
            $F[i] \leftarrow j + 1$
            $i \leftarrow i + 1$
            $j \leftarrow j + 1$
        else if $j > 0$ then
            {use failure function to shift $P$}
            $j \leftarrow F[j - 1]$
        else
            $F[i] \leftarrow 0$ { no match }
            $i \leftarrow i + 1$
```

Question # 11 **[1+3 =4 marks]**

a. Given the line segment $\overrightarrow{(a, b)}$ in below figure, design a brute force algorithm to determine whether this line will be the part of hull. Hint: use counterclockwise turn technique.



$$2 \times Area(a,b,c) = \begin{vmatrix} a_x & a_y & 1 \\ b_x & b_y & 1 \\ c_x & c_y & 1 \end{vmatrix} = (b_x - a_x)(c_y - a_y) - (b_y - a_y)(c_x - a_x)$$

- If area > 0 then a-b-c is counterclockwise.
- If area < 0, then a-b-c is clockwise.
- If area = 0, then a-b-c are collinear.

b. Show all the necessary computational steps carried out by your designed algorithm to determine the line is the part of hull.

Question # 12

Compute the time complexity of the both given algorithms, show the steps. **[2+2 =4 Marks]**

```
int main() {


        int i=1; int s=1;
        int n;
        cout<<"enter value of  N";
        cin>>n;
          while(s<n){
                i++;
                s=s+i;
                cout<<'*';
          }
  return 0;
}
```

```
int main() {

  for (int i =1; i<=n/3; i++){
          for (int j=1; j<n; j=j+4){

                 j=n;
                 while(j<=1){
                    cout<<j;
                     j=j/2;
                  }
          }
  }
  return 0;
}
```