National University of Computer & Emerging Sciences, Karachi
Fall-2020 Department of Computer Science
Assignment 5
Due: 25th December 2020

Max Marks: 60 Points

<u>Question # 1</u> 20 Points

Explain in your own words

(a) What is meant by P and NP Problems? Explain P = NP
(b) Why it is important to find approximate solutions for NP Complete Problems
(c) What is the difference between NP Complete and NP Hard
(d) A problem that is solvable in time complexity of $T(n) = 3 * n^n$ and space complexity of $S(n) = n^2$ and it can be validated in $T(n) = 2^n$ time. Is it a NP-Complete or NP-Hard? Explain

Question #2

Consider the following APPROX-VERTEX-COVER algorithm. Proof that this algorithm is 2-approximation method for VERTEX-COVER. 10 Points

```
APPROX-VERTEX-COVER(G)
    C = ∅;
    E'=G.E;
    while(E' ≠ ∅ ){
        Randomly choose a edge (u,v) in E', put u and v into C;
        Remove all the edges that covered by u or v from E'
    }
    Return C;
```

<u>Question 3</u> 10 Points

An Instance (X, F) of the set-covering problem consists of a finite set X and a family F of subset of X, such that every element of X belongs to at least one subset of F:

$$X = \bigcup_{S \in F} S$$

We say that a subset $S \in F$ covers all elements in X. Our goal is to find a minimum size subset $C \subseteq F$ whose members cover all of X.

$$X = \bigcup_{S \in C} S$$

Algorithm 1: GREEDY-SET-COVER (X, F)

$U \leftarrow X$
$C \leftarrow \emptyset$
While $U \neq 0$
  do select an $S \in F$ that maximizes $|S \cap U|$
    $U \leftarrow U - S$
    $C \leftarrow C \cup \{S\}$
return C

Consider each of the following words as a set of letters: {arid, dash, drain, heard, lost, nose, shun, slate, snare, thread}. Show which set cover GREEDY-SET-COVER produces, when we break ties in favor of the word that appears first in the dictionary.

Question 4: 20 Points

Consider following points in 2D
(6,2), (9,5), (-2,2), (-3,4), (-8,8), (-10,4), (-10,3), (-8,-6), (-4,-4), (6,4), (6,-6), (-6,-10), (8,0)
Find the smallest convex set containing all the points using Package Wrap (Jarvis March) and Graham Scan (Show all iterations).

BEST OF LUCK

National University of Computer & Emerging Sciences, Karachi
Fall-2020 Department of Computer Science
Assignment 5
Due: 25th December 2020

Max Marks: 60 Points

<u>Question # 1</u> 20 Points

Explain in your own words

(a) What is meant by P and NP Problems? Explain P = NP

P problems

a. (The original definition) Problems that can be solved by **deterministic Turing machine** in polynomial-time.
b. (A equivalent definition) Problems that are solvable in polynomial time.

NP problems

c. (The original definition) Problems that can be solved by **non-deterministic Turing machine** in polynomial-time.
d. (A equivalent definition) Problems that are **verifiable** in polynomial time.
   i. Given a solution, there is a polynomial-time algorithm to tell if this solution is correct.

P = NP means whether an NP problem can belong to class P problem. In other words, whether every problem whose solution can be verified by a computer in polynomial time can also be solved by a computer in polynomial tim

(b) Why it is important to find approximate solutions for NP Complete Problems
Sol: If a problem is NP-complete, there is very likely no polynomial-time algorithm to find an optimal solution. The idea of approximation algorithms is to develop polynomial-time algorithms to find a near optimal solution

(c) What is the difference between NP Complete and NP Hard
NP Problem:
The NP problems set of problems whose solutions are hard to find but easy to verify and are solved by Non-Deterministic Machine in polynomial time.
NP-Hard Problem:
Any decision problem P_i is called NP-Hard if and only if every problem of NP(say P<subj) is reducible to P_i in polynomial time.
NP-Complete Problem: Any problem is NP-Complete if it is a part of both NP and NP-Hard Problem.

(d) A problem that is solvable in time complexity of $T(n) = 3 * n^n$ and space complexity of $S(n) = n^2$ and it can be validated in $T(n) = 2^n$ time. Is it a NP-Complete or NP-Hard? Explain
Sol: NP-Hard as validated in $T(n) = 2^n$ time.

Question #2

Consider the following APPROX-VERTEX-COVER algorithm. Proof that this algorithm is 2-approximation method for VERTEX-COVER. 10 Points

```
APPROX-VERTEX-COVER(G)
    C = ∅;
    E'=G.E;
    while(E' ≠ ∅ ){
        Randomly choose a edge (u,v) in E', put u and v into C;
        Remove all the edges that covered by u or v from E'
    }
    Return C;
```

Proof:

- Assume a minimum vertex-cover is U*
- A vertex-cover produced by **APPROX-VERTEX-COVER**(G) is U
- The edges chosen in **APPROX-VERTEX-COVER**(G) is A
- A vertex in U* can only cover 1 edge in A
  - So |U*|>= |A|
- For each edge in A, there are 2 vertices in U
  - So |U|=2|A|
- So |U*|>= |U|/2
- So $\frac{|U|}{|U^*|} \leq 2$

<u>Question 3</u> 10 Points

An Instance (X, F) of the set-covering problem consists of a finite set X and a family F of subset of X, such that every element of X belongs to at least one subset of F:

$$X = \bigcup_{S \in F} S$$

We say that a subset $S \in F$ covers all elements in X. Our goal is to find a minimum size subset $C \subseteq F$ whose members cover all of X.

$$X = \bigcup_{S \in C} S$$

Algorithm 1: GREEDY-SET-COVER (X, F)

$U \leftarrow X$
$C \leftarrow \emptyset$
While $U \neq 0$
    do select an $S \in F$ that maximizes $|S \cap U|$
        $U \leftarrow U - S$
        $C \leftarrow C \cup \{S\}$
return C

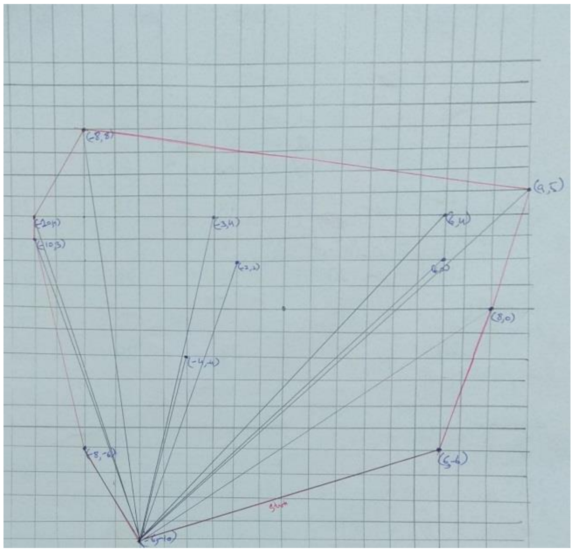
Consider each of the following words as a set of letters: {arid, dash, drain, heard, lost, nose, shun, slate, snare, thread}. Show which set cover GREEDY-SET-COVER produces, when we break ties in favor of the word that appears first in the dictionary

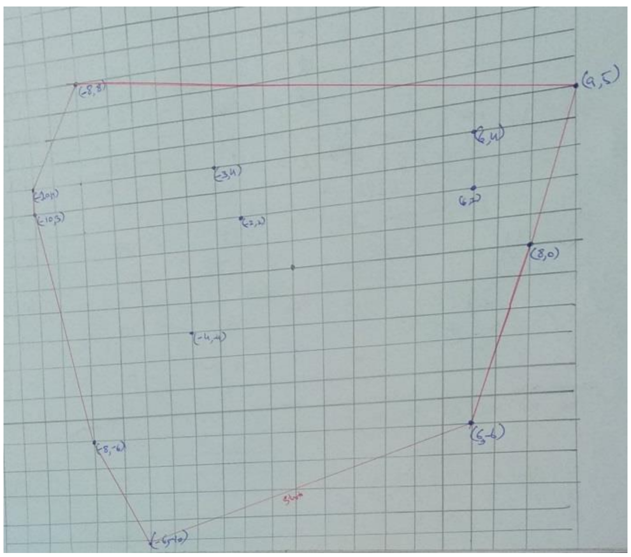
Since all of the words have no repeated letters, the first word selected will be the one that appears earliest on among those with the most letters, this is "thread". Now, we look among the words that are left, seeing how many letters that aren't already covered that they contain. Since "lost" has four letters that have not been mentioned yet,and it is first among those that do, that is the next one we select. The next one we pick is "drain" because it is has two unmentioned letters. This only leave "shun" having any unmentioned letters, so we pick that, completing our set. So, the final set of words in our cover is $\{thread, lost, drain, shun\}$.

Question 4: 20 Points

Consider following points in 2D
(6,2), (9,5), (-2,2), (-3,4), (-8,8), (-10,4), (-10,3), (-8,-6), (-4,-4), (6,4), (6,-6), (-6,-10), (8,0)
Find the smallest convex set containing all the points using Package Wrap (Jarvis March) and Graham Scan (Show all iterations)

A) Jarvis March: The points at the hull of convex in ordered form are: $(-6,-10) \rightarrow (6,-6) \rightarrow (8,0) \rightarrow (9,5) \rightarrow (-8,8) \rightarrow (-10,4) \rightarrow (-10,3) \rightarrow (-8,-6)$.
B) Graham Scan: The points are as same as the Jarvis March but the deleted points are: $(6,2) \rightarrow (6,4) \rightarrow (-2,2) \rightarrow (-4,-4) \rightarrow (-3,4)$. The sorted points are: $(-6,-10) \rightarrow (6,-6) \rightarrow (8,0) \rightarrow (9,5) \rightarrow (6,2) \rightarrow (6,4) \rightarrow (-2,2) \rightarrow (-4,-4) \rightarrow (-3,4) \rightarrow (-8,8) \rightarrow (-10,4) \rightarrow (-10,3) \rightarrow (-8,-6)$.


(-8,3)
(9,5)
(-3,4)
(6,4)
(8,0)
(-4,-4)
(-8,-6)
(5,-6)
(-5,-10)


(-8,3)
(9,5)
(6,4)
(-3,4)
(8,0)
(-4,-4)
(5,-6)
(-8,-6)
(-5,-10)