Thank you for taking the Week 3: Assignment 3.

Week 3: Assignment 3

Your last recorded submission was on 2023-02-14, 17:37 IST Due date: 2023-02-15, 23:59 IST.

1) Select the statement which is **not true** about Greedy algorithms. ***1 point***

- ○ It will never go back and change the previous decisions
- ◉ It will always eventually reach the global optimum
- ○ It is space conservative
- ○ Next decision is made from the current criterion

2) Meetings M1, M2, ..., M11 are to be conducted in a single available meeting room. The table below gives the `start` and `end` times of these meetings. If any meeting finishes at time `t`, then other meeting can be started at time `t` or afterwards.

What is the maximum number of meetings that can be held in the meeting room without conflicts?

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11
start	4	2	7	9	10	7	6	3	6	2	8
end	7	5	10	12	14	10	8	5	9	7	9

4

1 point

3) Which of the following greedy strategy would always work correctly for the problem scenario given in the previous question-2 ? ***1 point***

- ○ Always choose the meeting whose starting time is earliest and does not conflict with the previously selected meeting.
- ○ Always choose the meeting spanning the shortest interval and does not conflict with the previously selected meeting.
- ○ Always chose the meeting that overlaps the minimum number of other meetings and does not conflict with the previously selected meeting.
- ◉ Always choose the meeting whose end time is earliest and does not conflict with the previously selected meeting.

4) For an activity X, X_s is the starting time and X_e is the ending time. The following are the starting time and ending time of activities A, B, C, D, E, F, G and H given in chronological order.

$$A_s,\ B_s,\ C_s,\ A_e,\ D_s,\ C_e,\ E_s,\ B_e,\ F_s,\ D_e,\ G_s,\ E_e,\ H_s,\ F_e,\ G_e,\ H_e$$

What is the minimum number of rooms required to schedule the activities in a set of rooms such that there are no conflicts?

3

1 point

Question 5 & 6

Let `coins` be a list of `k` coin denominations (positive integers) that are available in as much quantity as you want. You want to pay a sum of Rupees `x` using as few coins as possible from the coin denominations given in the list `coins` .

Consider the following function **minimumCoins(coins, x)** to calculate the minimum number of coins

that can be used to pay the amount x .

```
def minimumCoins(coins, x):
  numCoins = 0
  coins.sort()
  while (x > 0):
    large = findLargestCoinLessThan_k(coins, x)
    if (large == None):
      return None
    else:
      x = x - large
      numCoins += 1
  return numCoins

def findLargestCoinLessThan_k(coins, k):
  i = len(coins) - 1
  while (i >= 0):
    if (coins[i] <= k):
      return coins[i]
    i -= 1
  return None
```

5) The function **minimumCoins(coins, x)** is an example of a greedy technique. ***1 point***

- (●) True
- ( ) False

6) Which of the following statements is/are true for the above solution. ***1 point***

- ( ) For coins=[6, 1, 5, 7] and x=15, the above function will return a wrong solution.
- ( ) For coins=[10, 1, 2, 5] and x=25, the above function will return a wrong solution.
- (●) For coins=[1, 2, 8, 9, 5, 10] and x=17, the above function will return a wrong solution.
- ( ) The above function will always return a correct solution.

7) Consider a random sequence of positive integers $x_1, x_2, x_3, \ldots, x_n$, we have to partition them in two sets **P** $(p_1, p_2, \ldots p_k)$ and **Q** $(q_1, q_2, \ldots q_{n-k})$ such that the difference between the sum of these two sets is minimum i.e. $|\sum p_i - \sum q_i|$ is minimum. ***1 point***

Which of the following greedy strategy would always work correctly for this problem?

- ( ) Sort $x_1, x_2, x_3, \ldots, x_n$ and put all even positioned elements in P and odd positioned element in Q from the sorted sequence.
- ( ) Find the median element of $x_1, x_2, x_3, \ldots, x_n$ put all elements lesser or equal to the median in P and greater than the median in Q.
- ( )

Assessment submitted.
X

Sort $x_1, x_2, x_3, \ldots, x_n$, from the sorted sequence at i^{th} step, x_i is placed in that set (P or Q) whose sum of all current elements is smaller in the i^{th} step.

- (●) None of the above

Question 8 ,9 & 10

Consider a problem scenario where a company has n jobs to be performed on a single machine one by one. Each job i has the time (T_i) required to complete and Deadline (D_i). Some jobs may complete after the deadline, so the lateness of each job is defined as follows:

$$L(i) = \left\{ \begin{array}{c} if\ \ TimeRequired(Ti) > Deadline(Di),\ \ TimeRequired(Ti) - Deadline(Di) \\ else, 0 \end{array} \right\}$$

Suppose M_L is the maximum lateness of all lateness:-

$$M_L = Max(L_1, L_2 \ldots L_n)$$

Here, the manager's goal is to minimize this M_L by scheduling the jobs in the appropriate order.

8) Consider the following jobs. ***1 point***

jobs ID	Time required(T)	Deadline(D)
1	4	6
2	2	7
3	1	8
4	4	12
5	3	14
6	2	5

What will be minimum possible value of M_L(maximum lateness)? Consider that start time of the first selected job is 0.

- ( ) 1
- (●) 2
- ( ) 3
- ( ) 4

9) Which of the following greedy strategy would always work correctly for the given problem scenario? ***1 point***

- ( ) Always choose the job first whose required time is minimum.
- (●) Always choose the job first whose deadline is earliest.
- ( ) Always choose the job first whose $D(Deadline) - T(RequiredTime)$ is minimum.
- ( ) None of the above

Assessment submitted.
X

10) A manager claims scheduling jobs based on the slack time to time taken ratio $\left(\frac{D(i)-T(i)}{T(i)}\right)$ is optimum. His strategy is to complete the job which has the least slack to the time taken ratio. Choose the counter-example to prove the strategy is not optimum. ***1 point***

- ○ T[1] = 2; D[1] = 15;
  T[2] = 4; D[2] = 9;
- ○ T[1] = 9; D[1] = 10;
  T[2] = 2; D[2] = 2;
- ○ T[1] = 1; D[1] = 10;
  T[2] = 2; D[2] = 3;
- ◉ T[1] = 9; D[1] = 15;
  T[2] = 4; D[2] = 9;

You may submit any number of times before the due date. The final submission will be considered for grading.

Submit Answers