

**CORSO DI  
ALGORITMI E STRUTTURE DATI**

**Prof. ROBERTO PIETRANTUONO**

**Prova del 17/01/2022**

**Indicazioni**

Si consegna un file in **formato editabile (.txt, .docx, .rtf, etc.)** nominandolo "*CognomeNome*", in cui è riportata l'implementazione (nel linguaggio scelto) seguita da una indicazione della complessità temporale dell'algoritmo implementato (complessità nel caso peggiore, è sufficiente il limite superiore  $O(f(n))$ ). Se si utilizzano librerie di cui non si conosce la complessità, lo si indichi nella spiegazione (ad esempio, "la complessità è  $O(n \log n)$  al netto della complessità dell'algoritmo  $x$ , che è non nota"). Se si utilizza la randomizzazione, si indichi anche il tempo di esecuzione atteso.

**PROBLEMA**

I traghetti fluviali servono sono usati per trasportare auto da una riva all'altra di un fiume. Le auto salgono sul traghetto da un'estremità, il traghetto attraversa il fiume e le auto escono dall'altra estremità del traghetto. Un traghetto può portare  $n$  auto in  $t$  minuti e tornare in  $t$  minuti.

Sia  $m$  il numero di auto che arrivano al terminal dei traghetti entro un determinato orario.

Qual è il minimo tempo entro cui tutte le auto possono essere trasportate attraverso il fiume? Qual è il numero minimo di viaggi che l'operatore deve effettuare per trasportare tutte le auto entro tale tempo?

**INPUT**

La prima riga di input contiene il numero di casi di test. Ogni test case inizia con  $n$ ,  $t$ ,  $m$ . Seguono  $m$  linee, ciascuna delle quali indica l'orario di arrivo di un'auto (in minuti dall'inizio della giornata).

**OUTPUT**

Per ogni test case, si stampi un'unica riga con due numeri interi: il tempo, in minuti dall'inizio della giornata, in cui l'ultima vettura viene consegnata sull'altra sponda del fiume, e il numero minimo di viaggi effettuati dal traghetto per trasportare le auto entro quel tempo.

Si può presumere che  $0 < n, t, m < 1440$ . I tempi di arrivo per ogni test case sono in ordine non decrescente.

**Sample Input**

```
2
2 10 10
0
10
20
30
40
50
60
70
80
90
```

2 10 3

10

30

40

### **Sample Output**

100 5

50 2