

---

## **Primo Progetto ASD 2018/2019**

**Cristian Consonni/Lorenzo Ghio**

# Montesor Va Alla Guerra

---

**Primo Progetto ASD 2018/2019**

**Cristian Consonni/Lorenzo Ghio**

## La Guerra di Montresor

Truppe nemiche hanno attaccato la Repubblica di Asdonia. Lo stato maggiore ha affidato la preparazione del contrattacco al generale Montresor.

Siete stati chiamati a raccolta per discutere del piano di contrattacco.

Il generale è noto per essere un impareggiabile condottiero. Le sue grandi abilità strategiche gli permettono di guidare le truppe spingendole a compiere incredibili ed eroiche gesta.

Questa volta però la missione è davvero difficile.



Il Generale Montresor  
(reperito fotografico)

## La missione

Si tratta di portare l'**arma finale** nella posizione di attacco. Il generale è in possesso infatti di un'arma in grado di determinare le sorti della guerra.



La potentissima arma ha però una corta gittata ed è molto ingombrante. Portarla nei pressi del campo nemico è davvero un'impresa!

## L'assemblaggio

Non solo, passare inosservati è fondamentale per cogliere il nemico di sorpresa. Il generale ha fatto allora smontare l'arma, poi le varie componenti sono state furtivamente paracadutate in un'area nei pressi del *target*, ovvero il luogo in cui il generale ha deciso di posizionare l'arma prima di scatenare l'offensiva.



## Un tempismo perfetto

Tutte le componenti hanno un transponder che permette al generale Montresor di conoscere l'esatta posizione dei vari pezzi sparpagliati nell'area.

Alcuni soldati sono nascosti e attendono l'ordine per scattare e raccogliere una delle componenti da portare al target.

Queste componenti sono molto pesanti: un soldato che corre è in grado di trasportare un solo pezzo per volta.

Bisogna agire in fretta, altrimenti il nemico potrebbe accorgersi della situazione e vanificare il contrattacco.



## Regole per la Missione (I)

- L'area di combattimento è mappata con delle coordinate cartesiane;
- Ognuna delle  $C$  componenti dell'arma si trova in un certo punto di quest'area, con coordinate  $(x_c, y_c)$ ;
- Similmente, ciascuno degli  $S$  soldati sul campo è acquattato nel suo nascondiglio iniziale di coordinate  $(x_s, y_s)$ ;

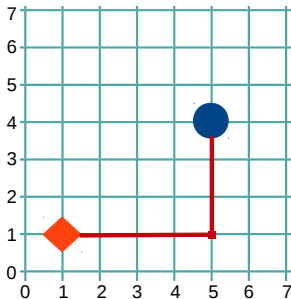
## Regole per la Missione (II)

- Data la conformazione del terreno, i soldati si spostano seguendo delle piste prestabilite. Per andare da un punto  $A = (x_A, y_A)$  ad un punto  $B = (x_B, y_B)$ , passeranno dal punto  $(x_A, y_B)$ :

distanza di Manhattan

$$d(A, B) = |x_A - x_B| + |y_A - y_B|$$

- Quando corrono, i soldati sono esposti al fuoco nemico. Il generale organizza dei diversivi e ordina ai soldati di correre **uno alla volta**.



Esempio: il soldato in  $(5, 4)$  per andare in  $(1, 1)$ , percorre una distanza pari a 7



## Regole per la Missione (III)

- Il terreno è accidentato, ma i soldati sono ben allenati: corrono sempre a tutta velocità. Percorrono un'unità di spazio per unità di tempo;
- Per recuperare una componente, un soldato impiegherà pertanto un tempo pari alla somma di due distanze: dal suo nascondiglio alla componente, e dalla componente al *target*;
- A seconda della necessità strategica, a un soldato può venire assegnato il recupero di 0, 1, o più componenti;
  - alcuni soldati possono non essere impiegati nella missione;
  - per recuperare 2 pezzi dell'arma un soldato dovrà andare dal suo nascondiglio fino al primo pezzo, poi fino al target (dove lascia il pezzo), quindi raggiungere il secondo pezzo e portarlo al target, dove rimane.

## Il problema

Il generale Montresor vi incarica di definire la strategia per il recupero di tutte le componenti. Conoscendo la disposizione delle componenti nell'area e l'ubicazione dei nascondigli dei soldati, dovete:

### Obiettivo Principale

Calcolare il tempo minimo per raccogliere tutti i pezzi e portarli nel punto prescelto.



## Medaglie al valore

Il generale Montresor è solito premiare i valorosi con onorificenze e medaglie. Per essere imparziale ed equo ha bisogno di sapere quali e quanti pezzi sono stati recuperati da ciascun soldato.

Oltre all'obiettivo principale, per completare la missione avete questo secondo obiettivo:

### Obiettivo Secondario

Consegnare al generale un rapporto completo in cui, per ogni componente, è associato il codice del soldato incaricato del recupero.



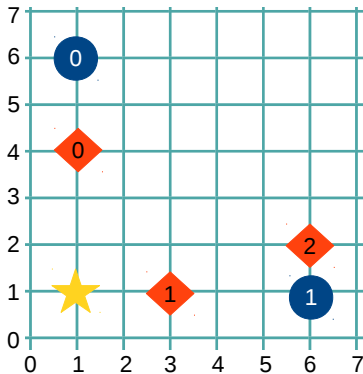
## Esempio

- Pallini blu: soldati
- Rombi rossi: componenti arma
- Stella gialla: target

Calcolo del tempo minimo:

1. Soldato 0 porta al target il pezzo 0.  
Tempo: 5
2. Soldato 1 recupera il pezzo 2 e lo porta al target. Tempo: 7
3. Uno dei soldati già al target recupera il pezzo 1 rimasto in campo. Tempo: 4

⇒ Totale:  $5 + 7 + 4 = 16$



## Input/Output

**Input:** Un file con  $1 + C + S + 1$  righe:

- La prima riga riporta 2 numeri interi: il numero di componenti  $C$  e di soldati  $S$ . Componenti e soldati sono numerati progressivamente a partire da 0;
- Le successive  $C$  righe riportano le coordinate di ogni componente: 2 numeri interi separati da spazio;
- Le successive  $S$  righe riportano similmente le coordinate dei soldati;
- L'ultima riga riporta le coordinate del *target*.

**Output:**

- Un numero intero: il tempo minimo  $T$  per completare la missione;
- **[opzionale, ma vedete il punteggio]**  $C$  righe, con un numero intero per riga. L' $i$ -esima di queste righe indica l'id del soldato preposto al recupero dell' $i$ -esimo componente.

## Esempio I/O (solo tempo)

INPUT

3 2

1 4

3 1

6 2

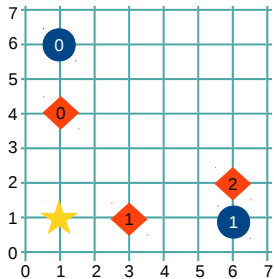
1 6

6 1

1 1

OUTPUT

16



## Esempio I/O (completo)

INPUT

3 2

1 4

3 1

6 2

1 6

6 1

1 1

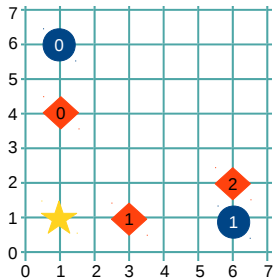
OUTPUT

16

0

1

1



# Assunzioni e note

## Assunzioni

- $1 \leq C, S \leq 1500$
- $\forall (x_i, y_i), -1000 \leq x_i, y_i \leq 1000$  (ovvero per componenti, soldati e *target*)

## Note

- Più soldati o componenti possono trovarsi nella stessa posizione, incluso il target.
- Può succedere che  $C < S$ ,  $C = S$ , o  $C > S$ .



# Test

## Casi di test

- 20 casi di test in totale;
- In almeno 6 casi c'è 1 solo soldato;
- In almeno 15 casi  $C, S \leq 500$ ;
- per superare il progetto - e sbloccare l'accesso all'orale  $\geq 30$  punti

## Dataset di esempio

- Dataset di esempio (con input e output) su

[https://judge.science.unitn.it/slides/asd18/dataset\\_guerra.zip](https://judge.science.unitn.it/slides/asd18/dataset_guerra.zip)

## Punteggi

Ogni caso di test vale 5 punti. Il punteggio massimo è di 100 punti.

Per ogni caso di test per cui la vostra soluzione fornisce un output entro i limiti di tempo e memoria:

- calcolare correttamente il tempo minimo  $T$  e fornire al contempo un *rapporto corretto* vale **5 punti**;
- calcolare correttamente solo il tempo minimo fa guadagnare **3 punti**;
- se  $T$  è errato: **0 punti**;
- se  $T$  è corretto, ma il rapporto è errato: **0 punti**.

⇒ con  $T$  errato (**non minimo**) si prendono sempre **0 punti**.

\**rapporto corretto*: vedere la prossima slide a proposito del correttore.

# Correttore

Come verifichiamo la correttezza delle vostre risposte?

## Tempo minimo

Confronto con il valore ottimo calcolato con la soluzione di riferimento.

## Correttezza rapporto

Possono esistere più rapporti ammissibili. Pertanto verificheremo che – per come avete assegnato i soldati alle varie componenti – sia veramente possibile ricondurre tutte le componenti al target nel tempo che avete calcolato.

## Esempio output alternativi

INPUT

3 2

1 4

3 1

6 2

1 6

6 1

1 1

OUTPUT

16

0

1

1

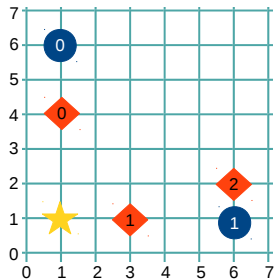
OUTPUT

16

0

0

1



# Consegna

## Consegna

giovedì 13 dicembre ore 18:00

- Per caricare il vostro codice, recatevi su <https://judge.science.unitn.it/arena/>

Avete poco più di una settimana di tempo, se avete difficoltà non esitate a chiederci aiuto!

- potete venire a ricevimento
- risponderemo alle vostre mail

... ma non durante il week-end :)

## Valutazione

- Il progetto è superato con  $\geq 30$  punti;
- Il progetto darà da 1 a 3 punti bonus allo scritto;
- Con la soluzione minima ( $\geq 30$  punti) prendente almeno 1 punto bonus e sbloccate l'accesso all'orale;
- Il ranking è disponibile all'indirizzo:  
`https://judge.science.unitn.it/arena/ranking/`;
- Conta il punteggio dell'**ultimo sorgente** inviato al sistema;
- Limite di 40 sottoposizioni per gruppo;

## Dataset di esempio

- È un dataset generato in modo analogo a quello usato da **arena** per la valutazione dei sorgenti che potete usare per provare le vostre soluzioni in locale;
- 20 casi di input con i corrispondenti output, ma con solo  $T$ ;
- Scaricabile da:  
`https://judge.science.unitn.it/slides/asd18/dataset\_guerra.zip`
- **Non** usate il sistema di valutazione come compilatore!

## Do's

### È permesso:

1. Discutere all'interno del gruppo
2. Chiedere chiarimenti sul testo
3. Chiedere opinioni su soluzioni
4. Sfruttare codice fornito nei laboratori
5. Utilizzare pseudocodice da libri o Wikipedia
6. Richiedere aiuto (anche pesante) per la soluzione "minima"
7. Venire a ricevimento



## Dont's

È vietato:

1. Discutere con altri gruppi
2. Mettere il proprio codice su repository pubblici
3. Utilizzare codice scritto da altri
4. Condividere codice (abbiamo potenti mezzi!)

## Ricevimento

Giorno	Data	Orario
giovedì	06/12	14:00-16:00
venerdì	07/12	10:00-12:00
lunedì	10/12	14:00-16:00
martedì	11/12	15:30-17:30 ( <b>lab</b> )
mercoledì	12/12	10:00-12:00
giovedì	13/12	10:00-12:00

### AVVISATECI VIA MAIL

Prima di venire a ricevimento è **obbligatorio richiedere appuntamento via mail!** Mettete entrambi gli esercitatori tra i destinatari.

[lorenzo.ghiro@unitn.it](mailto:lorenzo.ghiro@unitn.it)    [cristian.consonni@unitn.it](mailto:cristian.consonni@unitn.it)