

Terzo Progetto ASD 2017/2018

Alpinocalypse Now



“Per gli alpini nulla è NP-completo”

Adunata degli alpini

Che grande orgoglio e che spettacolo!



Adunata degli alpini

Che grande orgoglio e che spettacolo! Quando gli alpini si riuniscono per l'*adunata*, migliaia di penne nere raggiungono la città prescelta arrivando da tutta Italia.



Adunata degli alpini

Che grande orgoglio e che spettacolo! Quando gli alpini si riuniscono per l'*adunata*, migliaia di penne nere raggiungono la città prescelta arrivando da tutta Italia.

Quest'anno siamo stati fieri di ospitare gli alpini proprio a Trento :)



La città in festa

Nei giorni dell'adunata in città si fa festa!



La città in festa

Nei giorni dell'adunata in città si fa festa!
Lungo le strade si incontrano tanti alpini con i quali si può
cantare...



La città in festa



...o fare un brindisi!

La stanchezza

L'ultimo giorno dell'adunata gli alpini sfilano per ore nelle vie principali.

La stanchezza

L'ultimo giorno dell'adunata gli alpini sfilano per ore nelle vie principali. Poverini, i più anziani non hanno lo sprint di un tempo, tanto che i festeggiamenti prolungati possono risultare faticosi...

La stanchezza

L'ultimo giorno dell'adunata gli alpini sfilano per ore nelle vie principali. Poverini, i più anzianotti non hanno lo sprint di un tempo, tanto che i festeggiamenti prolungati possono risultare faticosi...
...molto faticosi!



Il vostro compito

Aiutate gli alpini “affaticati” (e forse rintonati dai festeggiamenti) a non stancarsi girando in tondo per le strade della città!

- ▶ Bisogna chiudere alcuni incroci stradali;
- ▶ È bene chiuderne il **meno possibile** per non bloccare la città;
- ▶ Dopo aver chiuso alcuni incroci, non devono esistere percorsi ciclici in cui gli alpini possano girare in tondo stancandosi inutilmente.

Input/Output

Input: un file contenente una lista di strade.

- ▶ La prima riga riporta 2 numeri, rispettivamente il numero di incroci e di strade;
- ▶ Ogni riga successiva é formata da 2 numeri e rappresenta una strada;
 - ▶ Questi numeri identificano gli incroci collegati dalla strada in questione.

Output: Il numero di incroci da chiudere seguito dalla lista (possibilmente minimale) di incroci da chiudere per non avere percorsi ciclici.

Esempio

INPUT

4 4

0 1

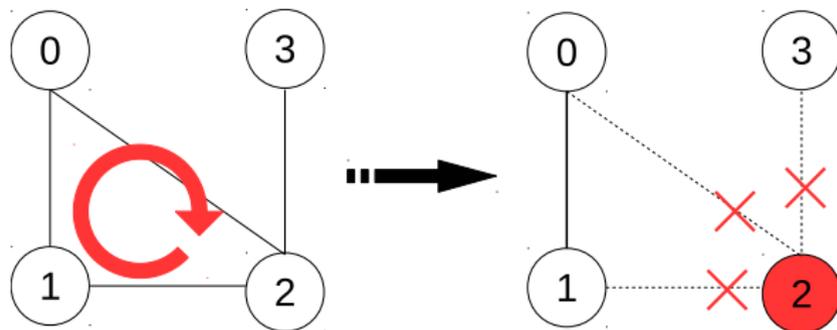
1 2

2 0

2 3

OUTPUT

1 2#



Note su Input

Limiti generali

- ▶ Il numero di incroci (I) è sempre inferiore a 100.
 - ▶ $3 \leq I < 100$

Il numero di strade S invece:

- ▶ In 3 casi di test su 10:
 - ▶ $S \leq 100$
- ▶ In 6 casi di test su 10:
 - ▶ $S \leq 500$
- ▶ In generale:
 - ▶ $S \leq 2000$

Output

Se scrivete una soluzione esponenziale (tipo branch and bound):

- ▶ Importate `alpini.h` (scaricabile da judge) e chiamate `helpers::setup()` come prima istruzione del main
- ▶ Man mano che migliorate la soluzione, scrivetela in output terminando la riga con `#`
- ▶ La libreria arresterà il programma prima del timeout

```
#include "alpini.h"
int main() {
    helpers::setup();
    ...
}
```

Il correttore considererà l'ultima riga di output che finisce con `#` quindi, anche se non appendete soluzioni multiple, terminate l'output con `#`.

Note su Output e dataset di esempio

Attenzione! Per gli input forniti nel dataset di esempio non è stata calcolata una soluzione ottima. Per questo motivo il dataset non contiene anche i relativi output, solitamente messi a disposizione. Gli output mostrati finora sono forniti a puro titolo illustrativo: rispettano la sintassi richiesta dal correttore ma non rappresentano necessariamente la soluzione migliore per il relativo input.

Punteggio

- ▶ 10 casi di test: ogni test assegna un punteggio (max 10 punti), che considera fino ai centesimi;
- ▶ Una soluzione proposta è valida se rende il grafo aciclico. Le risposte non valide non fanno punti!
- ▶ Le risposte valide invece ottengono questo punteggio:

$$P(S) = (1 - R/I) \times 10 + 0.1$$

NB: minore sarà il numero di nodi R rimossi, maggiore sarà il vostro punteggio!

- ▶ La **sufficienza è posta a 1 punto**. Vuol dire che avete perlomeno sempre fornito risposte valide;
- ▶ Non consideriamo soluzioni costanti. Non vale scrivere in output sempre la stessa cosa.

Calcolo del punteggio complessivo

Siamo arrivati alla fine dei laboratori, è ora di tirare le somme! I punti bonus verranno assegnati in questo modo:

- ▶ Sia $P2_{max}$ il miglior punteggio (da 0 a 100) registrato al progetto 2;
- ▶ Sia $P3_{max}$ l'analogo punteggio per il progetto 3;
- ▶ Sia Pk_i il punteggio ottenuto dal gruppo i nel progetto k (con $k \in \{2, 3\}$)

Il punteggio finale per il generico gruppo i lo calcoleremo così:

$$\text{Punteggio}(\text{gruppo}_i) = (P2_i/P2_{max} + P3_i/P3_{max})/2 \times 100$$

Punti bonus per l'esame

L'assegnazione punti avviene in maniera competitiva, come per il primo progetto:

- ▶ **3 punti** ai gruppi nel primo terzile della classifica (primo terzo della classifica, punteggio maggiore a quello fatto da almeno 2/3 dei gruppi);
- ▶ **2 punti** ai gruppi nel secondo terzile della classifica (secondo terzo della classifica);
- ▶ **1 punto** ai gruppi nel terzo terzile della classifica (ultimo terzo della classifica);

Consegna

Consegna mercoledì 30 maggio ore 23:59

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

Suggerimenti

Avete 10 giorni di tempo, cominciate subito a lavorare al progetto per presentarvi ai 2 prossimi laboratori con tutte le domande che vorrete fare.

Come al solito sappiate che:

- ▶ potete venire a ricevimento
- ▶ risponderemo alle vostre mail

...ma non durante il week-end!

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

Mi trovate nella **stanza 248, RS LAB, a Povo 1**, nei seguenti giorni e orari:

- ▶ Lunedì e giovedì
 - ▶ 14:00 - 16:00
 - ▶ Martedì, mercoledì, e venerdì
 - ▶ 10:00 - 12:00; 14:00 - 16:00
- ⇒ Prima di venire a ricevimento, è obbligatorio richiedere appuntamento via mail (lorenzo.ghiro@unitn.it)