

Indice

1	Panoramica introduttiva	1
2	Struttura costitutiva	1
2.1	Costruzione griglia	1
2.2	Percorsi	2
3	Operazioni eseguibili	3
3.1	Spunto di ottimizzazione	4

1 Panoramica introduttiva

Lo scopo del programma è quello di identificare dei percorsi minimi all'interno di una griglia esagonale. Questa griglia mima possibili percorsi, via terra o aria, di mezzi che devono trovare il modo più efficiente per spostarsi da un punto A ad un punto B. Le rotte possono essere modificate, editandone il costo, eliminate o aggiunte.

Infine, l'input al programma verrà pompato tramite `stdin`. Si può simulare questo pompaggio tramite un file, con il comando:

```
./program < input.txt
```

Per salvare anche l'output:

```
./program < input.txt > output.txt
```

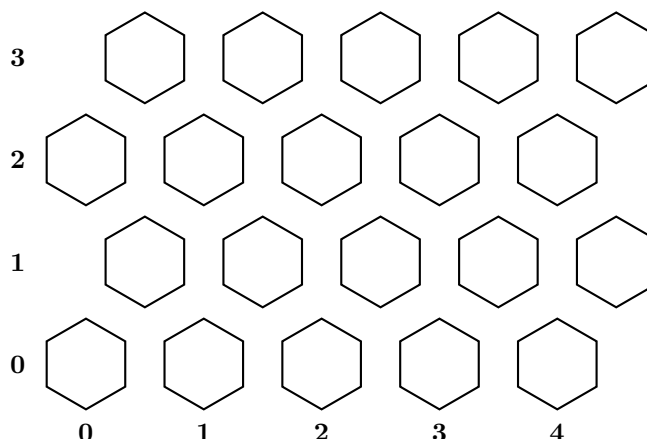
2 Struttura costitutiva

2.1 Costruzione griglia

La griglia è esagonale di forma rettangolare, di dimensione fissata all'inizio del programma. Ogni esagono è identificato dalle coordinate

(column, row)

Un esempio di griglia è la seguente:



Si nota però che con questo sistema di coordinate, chiamato *offset odd-r*:

- *Righe pari*: risultano ben allineate a sinistra
- *Righe dispari*: risultano "spostate" di mezzo esagono a destra

Non permette le basilari operazioni vettoriali quali somma, sottrazione, moltiplicazioni per uno scalare ...

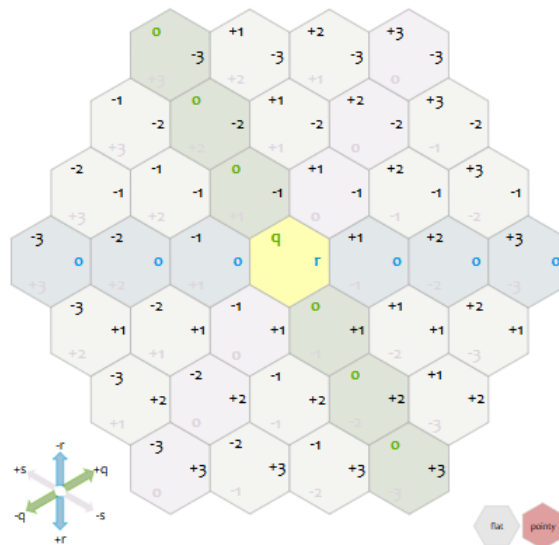
È utile, per calcolare le distanze, passare da coordinate esagonali a **coordinate assiali**. Queste si compongono di:

- **Row**: è la stessa delle coordinate esagonali

$$r = \text{row}$$

- **Oblique-Column**: rappresenta la posizione lungo una direzione obliqua che attraversa la griglia da nord-ovest a sud-est

$$q = \text{column} - (\text{row} \gg 1)$$

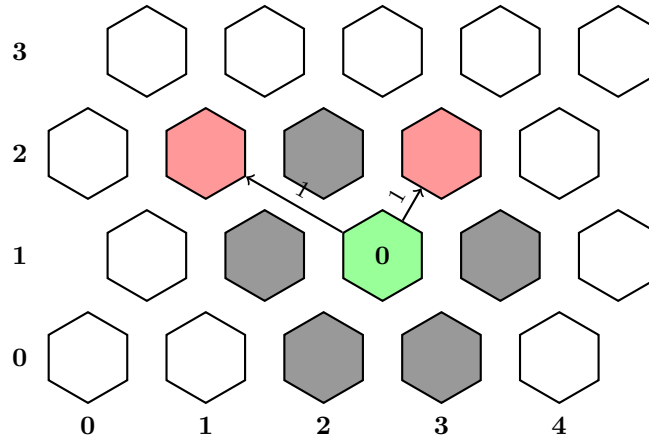


2.2 Percorsi

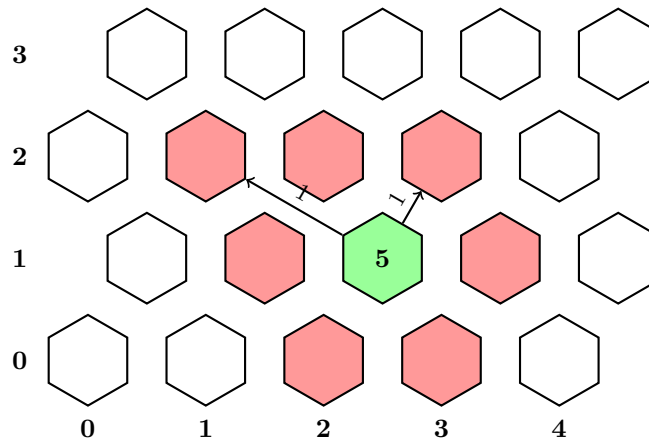
Ci sono due tipi di percorsi possibili:

- **Percorso via terra**: collega automaticamente un esagono con i vicini
 - **Esagono non abbandonabile**: ha costo 0 ed è un esagono da cui non si può partire via terra. Si può però *visitare*.
 - **Esagono abbandonabile**: ha un costo compreso tra 1-100
- **Percorso via aria**: collega un esagono con al più 5 altri esagoni, in qualsiasi posizione essi siano

Ogni percorso è un numero naturale (non negativo).



Esagono (1,2) non può essere lasciato via terra. Può però essere lasciato per via aerea



Costo via terra: 5
Costo via aerea: 1

3 Operazioni eseguibili

- `init < columns > < rows >`

Inizializza o reinizializza una nuova griglia di dimensione:

`rows × columns`

Stampando poi il messaggio: OK.

I parametri settati sono:

- Costo esagoni: 1
- Rotte aeree: nessuna

- `change_cost < column > < row > < param > < radius >`

Dove:

- `param`: nuovo parametro di costo, compreso tra `[-10, 10]`

- **radius**: raggio entro cui applicare il cambiamento

In particolare il raggio modifica ogni esagono a distanza:

$$d < \text{radius}$$

La modifica dei costi viene applicata al costo dell'**esagono** e di ogni **rotta** uscente secondo la formula:

$$\text{weight}_{(x_e, y_e)} = \text{weight}_{(x_e, y_e)} + \lfloor \langle \text{param} \rangle \times \max(0, \frac{\langle \text{radius} \rangle - \text{dist}(x_e, y_e, \langle x \rangle, \langle y \rangle)}{\langle \text{radius} \rangle}) \rfloor$$

Stampando poi:

- OK: se esagono valido
- KO: se l'esagono non è valido o $\langle \text{radius} \rangle = 0$

$\text{dist}(x_e, y_e, \langle x \rangle, \langle y \rangle)$ è il numero minimo di esagoni da *visitare* per arrivare a (x_e, y_e) ingorando i vari impedimenti (rotte, costi, non abbandonabilità), destinazione inclusa.

- **toggle_air_route** $\langle x1 \rangle \langle y1 \rangle \langle x2 \rangle \langle y2 \rangle$

Abbiamo 2 casi:

- Rotta già presente: viene rimossa
- Rotta non presente: viene aggiunta una rotta tra i due esagoni, dove il nuovo costo è la media per difetto di *tutte le connessioni uscenti dall'esagono 1 e del suo costo di uscita*.

Stampando poi:

- OK: se esagoni validi
- KO: se uno dei due esagoni non è valido o se l'esagono 1 dispone già di 5 rotte

- **travel_cost** $\langle x1 \rangle \langle y1 \rangle \langle x2 \rangle \langle y2 \rangle$

Calcola il costo minimo per raggiungere l'esagono 2 a partire dall'esagono 1, ignorando il costo dell'esagono 2.

Stampando poi:

- **weight**: il costo totale, differenziando via aria o via terra
- 0: se 1 e 2 coincidono, ovvero la destinazione coincide con la partenza
- -1: se se uno dei due esagoni non è valido o se non sono collegati in alcun modo

3.1 Spunto di ottimizzazione

- **Comandi poco usati**: **change_cost** e **toggle_air_route**

- **Comandi altamente usati**: **travel_cost**

Inoltre questo si concentra molto spesso su stesse zone della mappa: sorgente e destinazione sono spesso concentrate in una singola zona. Altre zone verranno completamente ignorate