

Struttura dati:

Tabella di simboli ST con campi: vettore di Vertex (struct contenente nome e grado di ogni vertice) e N (numero di vertici).

Grafo Graph con campi: due interi per memorizzare numero di vertici e di archi (rispettivamente V ed E), tabella di simboli tab (dichiarata esattamente come riportato poco fa), matrice di adiacenza allocata dinamicamente di dimensioni $V \times V$ (numeri di vertici) e vettore di archi di tipo Edge (struct contenente 2 interi che indicano la posizione nella tabella dei simboli dei vertici associati a un arco specifico).

Strategie risolutive:

Acquisizione grafo: lettura numero di vertici e acquisizione dei vertici standard, per la lettura degli archi, oltre ad aggiornare la matrice di adiacenza, salvo nel vettore di archi, gli archi stessi, inoltre aggiorno il grado di ogni vertice se esso è uno dei due vertici presenti in un arco (operazione effettuata su entrambi i vertici letti in ogni arco).

k_core: salvo in un vettore ausiliario allocato dinamicamente, di nome vertici, i gradi di ogni vertice. Avvio un for che va da 1 a k (inserito dall'utente) che, per ogni vertice, se esso ha grado $< k$ e maggiore di 0, chiama la funzione updateVertici (spiegata più avanti). Questa operazione viene effettuata k volte per non commettere errori di valutazione nell'eliminazione degli archi. Esempio: quanto specificato nel testo dell'esame con i vertici M N ed O.

updateVertici: funzione che dato un vertice, il vettore di vertici della funzione k_core e il grafo G, diminuisce i gradi (nel vettore vertici) del vertice inserito e di tutti i suoi (eventuali) vertici collegati.

Differenze:

-Implementazione di ST e Graph fatta come ADT di 1 tipo invece che come quasi ADT.

-Edgeinsert: aggiunti due campi (non avevo considerato che se voglio salvare un arco, devo pur sapere che arco è).

- STdiminuiscig non inserita, non era necessaria.
- Aggiunte le varie free (compresa STfree non dichiarata nel compito) e gli include delle librerie di base necessarie (stdio.h, stdlib.h, string.h).
- In Vertexinit aggiunti i “[i]” nella fase di inizializzazione.