

Prova scritta — appello di giugno 2022

In una cellula vivente avvengono continuamente reazioni di vario genere. Ogni reazione coinvolge diverse sostanze presenti nella cellula e produce nuove sostanze (ad esempio, proteine).

Possiamo dare un modello matematico di reazione in questo modo: fissiamo un insieme finito S di “sostanze”; una reazione è determinata da tre insiemi di sostanze:

- l'insieme R dei “reagenti” (*reactants*)
- l'insieme I degli “inibitori” (*inhibitors*)
- l'insieme P dei “prodotti” (*products*)

Descriviamo lo stato, o configurazione, della cellula in un certo istante specificando un sottoinsieme di S , $Q \subseteq S$. Gli elementi di Q sono le sostanze presenti nella cellula in quell'istante. Una reazione $\alpha = (R, I, P)$ è *abilitata* in Q se tutti i suoi reagenti sono presenti in Q e nessun inibitore è presente in Q .

Una reazione abilitata “accade” e ha l'effetto di consumare i reagenti e di produrre i suoi prodotti, modificando lo stato della cellula. Se in una configurazione diverse reazioni sono abilitate, supporremo che avvengano tutte simultaneamente. L'effetto di un insieme di reazioni è l'unione degli effetti delle singole reazioni. La configurazione successiva sarà formata dall'unione dei loro prodotti.

Si chiede di scrivere un programma che simula l'evoluzione di un sistema di reazioni a partire da una configurazione iniziale. Per semplicità, supponiamo che ogni reazione abbia due reagenti, un inibitore, e due prodotti, che l'insieme di sostanze contenga N elementi, e che ci siano K reazioni, con N e K fissati in direttive `#define`. Le sostanze sono identificate dai numeri interi $0, 1, \dots, N-1$.

Il programma deve leggere da *standard input* la configurazione iniziale, nella forma di elenco di numeri interi sulla prima riga. Ogni riga successiva descrive una reazione, come nell'esempio qui sotto:

a 2 4 3 0 2

Il primo carattere della riga è l'identificatore della reazione; i numeri successivi corrispondono, nell'ordine, ai due reagenti, all'inibitore, e ai due prodotti.

Il programma deve compiere una simulazione calcolando, a partire dalla configurazione iniziale, quali reazioni sono abilitate e determinando la configurazione successiva; il numero di passi di simulazione da compiere dev'essere passato al programma come argomento sulla riga di comando.

Il programma deve stampare le configurazioni prodotte, una per riga, calcolare quante volte, nell'arco della simulazione, avviene ogni reazione, e stampare i valori del conteggio.

Un esempio di dati di ingresso e dei corrispondenti risultati sarà pubblicato sul sito del corso.

Il programma dev'essere adeguatamente commentato. La valutazione terrà conto della correttezza del codice e della sua qualità (leggibilità, chiarezza dei commenti, articolazione in funzioni). Il codice deve essere scritto in ANSI C, come descritto in B.W. Kernighan e D.M. Ritchie, *Il linguaggio C* (seconda edizione), o in Stephen G. Kochan, *Programmare in C*. Non è consentito l'uso di librerie o funzioni non descritte in questi libri (salvo, ovviamente, quelle sviluppate da voi). Non è consentito dichiarare vettori la cui dimensione sia specificata dal nome di una variabile (è invece possibile allocare dinamicamente lo spazio per un vettore o per una matrice la cui dimensione non sia nota a priori). Il programma verrà compilato con gcc e con le opzioni `-Wall -std=c99` e, se necessario, `-lm`. In caso di dubbio, rivolgetevi al docente per chiarimenti.

Dovete consegnare, entro le 14 del 6 giugno 2022 il codice sorgente, spedito per posta elettronica a luca.bernardinello@unimib.it; il *subject*, o *oggetto*, del messaggio deve contenere la sigla LABINF seguita da uno spazio e dal numero di matricola dello studente; il file contenente il programma dev'essere allegato al messaggio. La prima riga del codice sorgente dev'essere un commento contenente nome, cognome e numero di matricola dello studente. Il nome del file deve consistere nel numero di matricola dello studente con l'estensione `c` (ad esempio, `710532.c`).