Esercizi del 24/1

INFORMAZIONI IMPORTANTI

I programmi richiesti devono leggere l'input da un file input1, input2, ecc. e devono scrivere l'output su un file output1, output2, ecc. In ogni esercizio sarà descritto in dettaglio cosa vada assunto sul contenuto del file di input e cosa vada scritto su quello di output. Il file che contiene la soluzione dell'esercizio (1) deve chiamarsi esercizio1.cpp e quello del (2) deve chiamarsi esercizio2.cpp. Per ogni esercizio trovate nella vostra home un file di input ed un corrispondente file di output. Si tratta solo di esempi. Dovreste provare i vostri programmi anche con altri file di input. I vostri programmi dovrebbero compilare e per gli input dati dovrebbero produrre output uguali a quelli dati.

ESERCIZI: Dovete cercare di scrivere (come commenti dei programmi) pre- e postcondizioni per ogni programma e invarianti per ogni ciclo.

1) Il main deve dichiarare un array int A[100] e leggere da input1 un intero dim che specifica quanti elementi inserire in A (certamente 1<=dim<=100), dopo di che il main deve leggere i dim interi che seguono su input1, inserendo i valori in A[0], A[1], ..., A[dim-1]. Dopo il main deve leggere da input1 due ulteriori interi che chiameremo x e y, con 0<=x<=dim. Si deve assumere che input1 contenga tutti gli interi richiesti e cioè: 1 valore per dim, dim valori da mettere in A e infine 2 valori per x e y. Inoltre si deve assumere che 1<=dim<=100 e 0<=x<=dim. Sarebbe quindi un errore se main controllasse che input1 contenga questi valori.

Dopo le operazioni di input appena descritte, il main deve stabilire se A[0..dim-1] contiene esattamente x occorrenze del valore y. Nel caso A contenga esattamente x istanze di y, allora main deve stampare su output1 il valore true e le x posizioni di A che contengono y (separate da uno spazio). Se invece A non soddisfa la condizione, main deve stampare su output1 il valore false e il numero di occorrenze di y in A[0..dim-1].

**Esempio**: *dim=5, A[0,3,-1,-1,2], x=0 e y=1, in output1 va scritto true e basta. Se invece x=2 e y=-1, allora su output1 va scritto true seguito da 2 e 3. Se x=1 e y=-1, su output1 va scritto false seguito da 2, visto che ci sono 2 occorrenze di -1 in A (e non 1).*

2)Dato un array A[0..dim] di interi, una sottosequenza crescente di A è una porzione A[i..j], con 0<=i<=j<=dim-1 tale che k in [i..j-1], A[k]<=A[k+1].

Si tratta di scrivere un main() che dichiari l'array int A[100] e poi, come nell'esercizio (1), legga da input2 un intero dim (1<=dim<=100) e dopo legga da input2 dim interi che andranno messi nelle prime dim posizioni di A. Dopo di che deve calcolare la sottosequenza crescente di lunghezza massima presente in A[0..dim-1]. Il main dovrà scrivere su output2, la coppia di indici che identifica l' inizio e la fine della sottosequenza crescente massima trovata. In caso ci siano diverse sottosequenze crescenti di uguale lunghezza, si deve scrivere su output2 l'inizio e la fine della sottosequenza crescente massima che ha inizio minimo.

**Esempio**: *sia dim=10 e A=[3,2,4,4,2,3,3,10,3,5]. La sottosequenza crescente massima è 2,3,3,10, che inizia nella posizione 4 di A e termina in posizione 7. Quindi si deve stampare 4 e 7. Se cambiamo il 10 di A in -1, allora avremmo 2 sottosequenze crescenti massime: 2,4,4, poi 2,3,3 e infine -1,3,5. Quindi si deve stampare inizio e fine della prima e cioè: 1 e 3.*