

Algoritmi e Strutture Dati (IN110)

Esercitazione n. 5

Marco Liverani *

Esercizio n. 1

Leggere in input una sequenza di n numeri interi e memorizzarla in un array A . Si supponga che la sequenza letta in input sia già ordinata in ordine crescente. Generare in modo casuale una seconda sequenza di m numeri interi ed inserire gli elementi generati nella posizione corretta nell'array A in modo che A continui ad essere ordinato, eventualmente spostando in avanti gli elementi già presenti per fare posto ai nuovi elementi da inserire. Stampare in output l'array A .

Esempio Sia $n = 8$ e sia $A = (3, 4, 7, 8, 17, 35)$ la sequenza letta in input. Sia $B = (15, 8, 6, 47)$ la sequenza di m interi generata in modo casuale. Allora al termine dell'elaborazione l'array A conterrà i seguenti elementi: $A = (3, 4, 6, 7, 8, 8, 15, 17, 35, 47)$.

Codifica in linguaggio C

```
1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <time.h>
4 #define MAX 100
5
6 int leggi_array(int V[]) {
7     int i, n;
8     printf("Numero di elementi: ");
9     scanf("%d", &n);
10    for (i=0; i<n; i++)
11        scanf("%d", &V[i]);
12    return(n);
13 }
14
15 int genera_array(int V[]) {
16     int i, n;
17     printf("Numero di elementi: ");
18     scanf("%d", &n);
19     srand((unsigned)time(NULL));
20     for (i=0; i<n; i++)
21         V[i] = rand()%100;
22     return(n);
23 }
```

*Università degli Studi Roma Tre, Corso di Laurea in Matematica, Corso di Algoritmi e Strutture Dati (IN110) – sito web del corso <http://www.mat.uniroma3.it/users/liverani/IN110/>

```

24
25 void stampa_array(int V[], int n) {
26     int i;
27     for (i=0; i<n; i++)
28         printf("%d ", V[i]);
29     printf("\n");
30     return;
31 }
32
33 void inserisci(int A[], int n, int B[], int m) {
34     int i, j;
35     for (j=0; j<m; j++) {
36         i = n+j-1;
37         while (i>=0 && A[i]>B[j]) {
38             A[i+1] = A[i];
39             i--;
40         }
41         A[i+1] = B[j];
42     }
43     return;
44 }
45
46 int main(void) {
47     int i, n, m, x, A[MAX], B[MAX];
48     n = leggi_array(A);
49     m = genera_array(B);
50     printf("La sequenza generata in modo casuale e':\n");
51     stampa_array(B, m);
52     inserisci(A, n, B, m);
53     stampa_array(A, n+m);
54     return(1);
55 }

```

Esercizio n. 2

Leggere in input due numeri floating point a e b ($a < b$) e il numero intero positivo n . Individuare n punti equidistanti nell'intervallo $[a, b]$, $x_0 = a, x_1, x_2, \dots, x_{n-1} = b$ e calcolare il valore della funzione $f(x) = \frac{1}{2}x^2 + 10\sin x$ negli n punti dell'intervallo. Salvare le coordinate dei punti su un file e disegnare il grafico utilizzando il programma Gnuplot e il comando "plot *filename*" with lines".

Codifica in linguaggio C

```
1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <math.h>
4 #define MAX 100
5
6 float f(float x) {
7     return(10*sin(x)+0.5*x*x);
8 }
9
10 int main(void) {
11     float a, b, delta, x, y;
12     int n;
13     char nome[MAX];
14     FILE *out;
15
16     printf("Inserisci gli estremi dell'intervallo [a,b]: ");
17     scanf("%f %f", &a, &b);
18     printf("Inserisci il numero di punti: ");
19     scanf("%d", &n);
20     printf("inserisci il nome del file: ");
21     scanf("%s", nome);
22     if ((out = fopen(nome, "w+t"))) {
23         delta = (b-a)/(n-1);
24         x = a;
25         while (x<=b) {
26             y = f(x);
27             fprintf(out, "%f %f\n", x, y);
28             x = x+delta;
29         }
30         fclose(out);
31         printf("Calcolo completato. Con Gnuplot puoi disegnare il grafico ");
32         printf("con il comando 'plot \"%s\" with lines'.\n", nome);
33     } else {
34         printf("ERRORE: impossibile aprire il file in scrittura!\n\n");
35     }
36     return(1);
37 }
```