

Prima prova di esonero – 2 novembre 2015

La prova dura tre ore, durante le quali non è possibile allontanarsi dall'aula, se non dopo aver consegnato l'elaborato scritto. Per superare la prova di esonero e poter sostenere la successiva è necessario ottenere almeno 15 punti. È possibile utilizzare libri e appunti personali, senza scambiarli con altri studenti. I compiti che presenteranno evidenti ed anomale "similitudini" saranno annullati.

Deve essere consegnata solo la "bella copia" del compito scritto; su ciascun foglio deve essere riportato il nome, il cognome e il numero di matricola (o un altro codice identificativo di fantasia) dello studente.

Esercizio n. 1 (max 10 punti)

Risolvere il seguente problema proponendo una pseudo-codifica dell'algoritmo, il diagramma di flusso ed infine la codifica in linguaggio C del programma che implementa l'algoritmo stesso.

Generare un array A di n numeri casuali compresi nell'insieme $\{50, 51, 52, \dots, 99, 100\}$. Costruire un array B i cui elementi b_i (con $i = 0, \dots, n - 1$) siano dati da a_i moltiplicato per il numero di occorrenze di a_i in A . Stampare A e B .

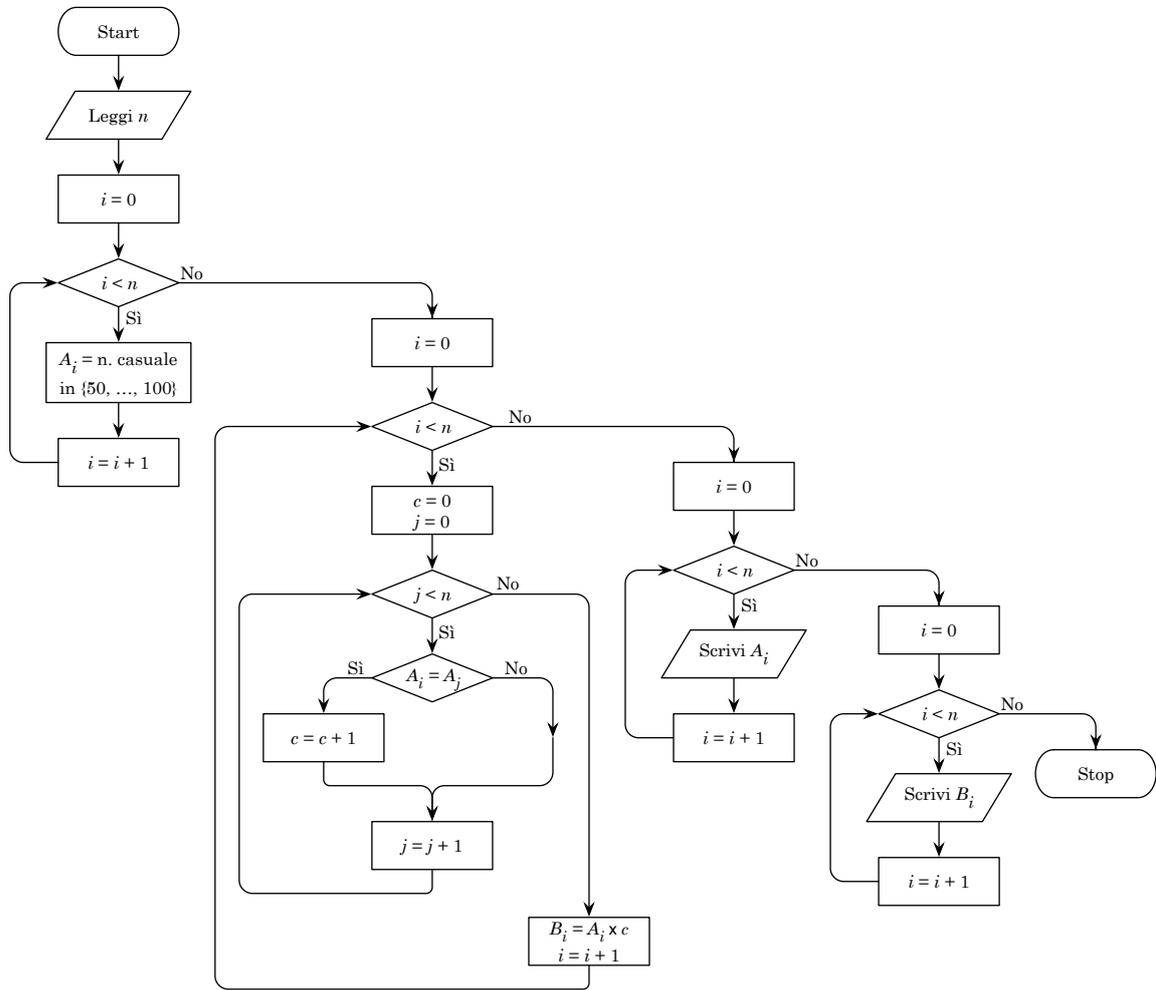
Esempio Supponiamo che $n = 10$ e $A = (63, 52, 74, 92, 100, 52, 99, 100, 52, 74)$. Allora il vettore B sarà costruito come segue: $B = (63 \times 1 = 63, 52 \times 3 = 156, 74 \times 2 = 148, 92 \times 1 = 92, 100 \times 2 = 200, 52 \times 3 = 156, 99 \times 1 = 99, 100 \times 2 = 200, 52 \times 3 = 156, 74 \times 2 = 148)$.

Soluzione

Pseudo-codifica dell'algoritmo

```
1: leggi  $n$ 
2: per  $i = 0, 1, \dots, n - 1$  ripeti:
3:    $a_i =$  numero casuale in  $\{50, \dots, 100\}$ 
4: fine-ciclo
5: per  $i = 0, 1, \dots, n - 1$  ripeti:
6:    $c = 0$ 
7:   per  $j = 0, 1, \dots, n - 1$  ripeti:
8:     se  $a_i = a_j$  allora
9:        $c = c + 1$ 
10:    fine-condizione
11:  fine-ciclo
12:   $b_i = a_i \times c$ 
13: fine-ciclo
14: per  $i = 0, 1, \dots, n - 1$  ripeti:
15:  scrivi  $a_i$ 
16: fine-ciclo
17: per  $i = 0, 1, \dots, n - 1$  ripeti:
18:  scrivi  $b_i$ 
19: fine-ciclo
20: stop
```

Diagramma di flusso



Codifica in linguaggio C

```
1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <time.h>
4 #define MAX 100
5
6 int generaArray(int X[]) {
7     int i, n;
8     printf("Numero di elementi: ");
9     scanf("%d", &n);
10    srand((unsigned)time(NULL));
11    for (i=0; i<n; i++)
12        X[i] = rand() % 51 + 50;
13    return(n);
14 }
15
16 void stampaArray(int X[], int n) {
17     int i;
18     for (i=0; i<n; i++)
19         printf("%4d ", X[i]);
20     printf("\n");
21     return;
22 }
23
24 void costruisciArray(int X[], int Y[], int n) {
25     int i, j, c;
26     for (i=0; i<n; i++) {
27         c = 0;
28         for (j = 0; j<n; j++)
29             if (X[i] == X[j])
30                 c++;
31         Y[i] = X[i] * c;
32     }
33     return;
34 }
35
36 int main(void) {
37     int A[MAX], B[MAX], n;
38     n = generaArray(A);
39     costruisciArray(A, B, n);
40     stampaArray(A, n);
41     stampaArray(B, n);
42     return(0);
43 }
```

Esercizio n. 2 (max 8 punti)

Scrivere il diagramma di flusso e la codifica in linguaggio C (un programma completo) del seguente algoritmo:

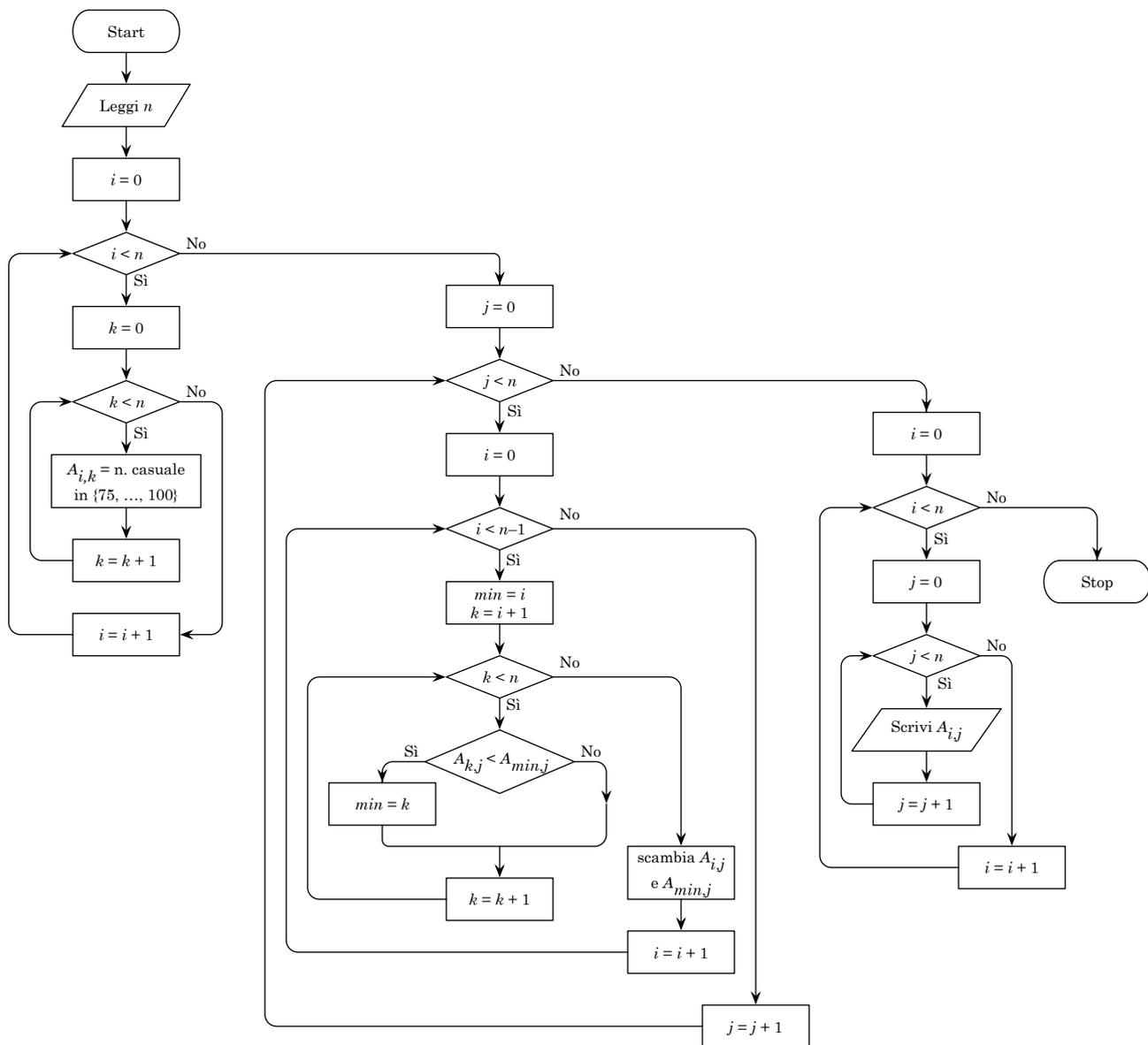
```
1: leggi  $n$ 
2: per  $i = 0, 1, \dots, n - 1$  ripeti:
3:    $k = 0$ 
4:   fintanto che  $k < n$  ripeti:
5:      $A_{i,k} =$  numero casuale compreso tra 75 e 100
6:      $k = k + 1$ 
7:   fine-ciclo
8: fine-ciclo
9: per  $j = 0, 1, \dots, n - 1$  ripeti:
10:  per  $i = 0, 1, \dots, n - 2$  ripeti:
11:    $min = i$ 
12:   per  $k = i + 1, \dots, n - 1$  ripeti:
13:    se  $A_{k,j} < A_{min,j}$  allora
14:      $min = k$ 
15:   fine-condizione
16:  fine-ciclo
17:  scambia  $A_{i,j}$  e  $A_{min,j}$ 
18: fine-ciclo
19: fine-ciclo
20: per  $i = 0, 1, \dots, n - 1$  ripeti:
21:  per  $j = 0, 1, \dots, n - 1$  ripeti:
22:   scrivi  $A_{i,j}$ 
23:  fine-ciclo
24: fine-ciclo
25: stop
```

Soluzione

Codifica in linguaggio C

```
1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <time.h>
4 #define MAX 100
5
6 int main(void) {
7     int A[MAX][MAX], i, j, k, n, min, x;
8     srand((unsigned)time(NULL));
9     scanf("%d", &n);
10    for (i=0; i<n; i++)
11        for (k=0; k<n; k++)
12            A[i][k] = rand()%26 + 75;
13    for (j=0; j<n; j++) {
14        for (i=0; i<n-1; i++) {
15            min = i;
16            for (k=i+1; k<n; k++)
17                if (A[k][j] < A[min][j])
18                    min = k;
19            x = A[i][j];
20            A[i][j] = A[min][j];
21            A[min][j] = x;
22        }
23    }
24    for (i=0; i<n; i++) {
25        for (j=0; j<n; j++)
26            printf("%3d ", A[i][j]);
27        printf("\n");
28    }
29    return(0);
30 }
```

Diagramma di flusso

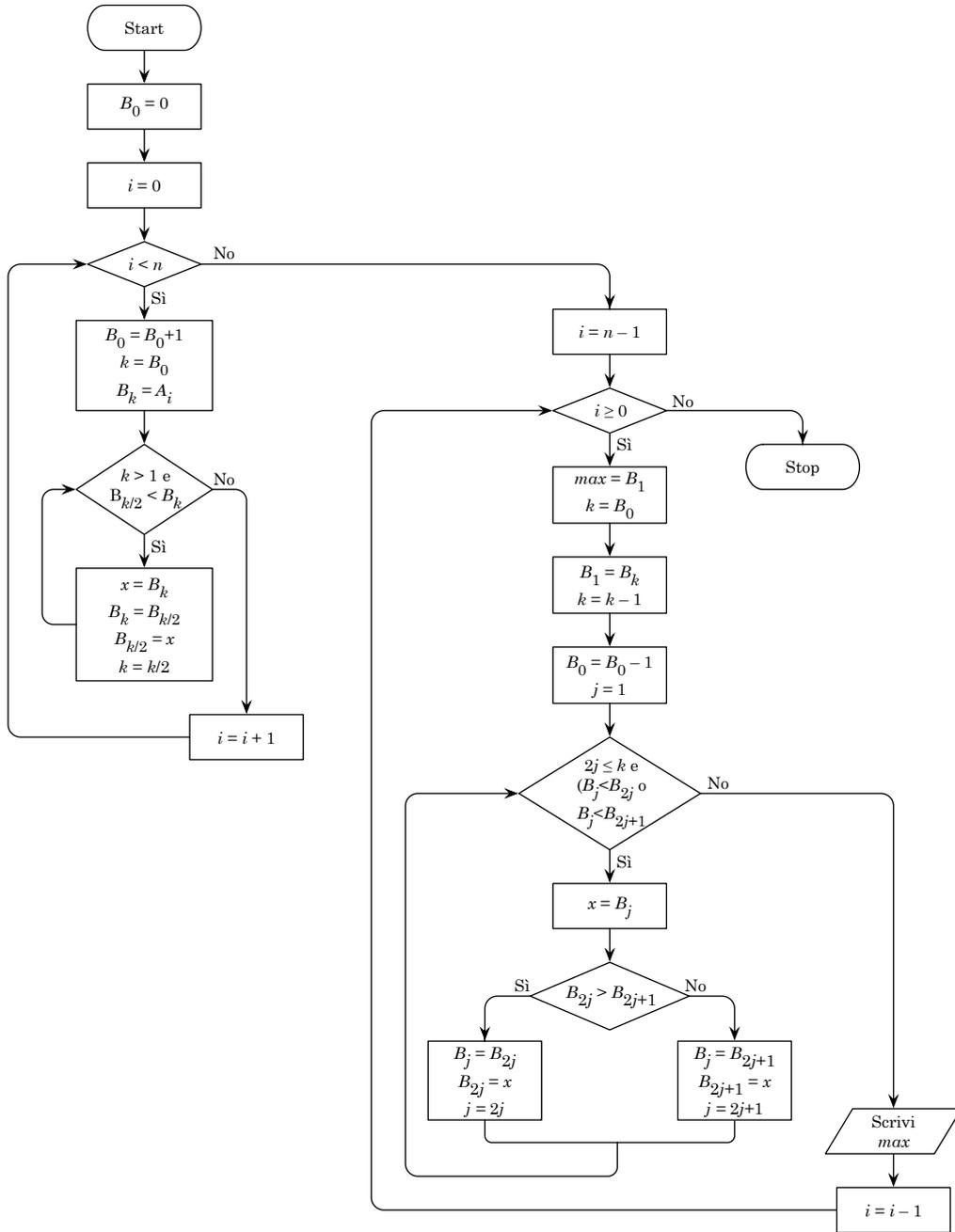


Esercizio n. 3 (max 6 punti)

Utilizzando le regole della programmazione strutturata, disegnare il diagramma di flusso della seguente funzione codificata in linguaggio C.

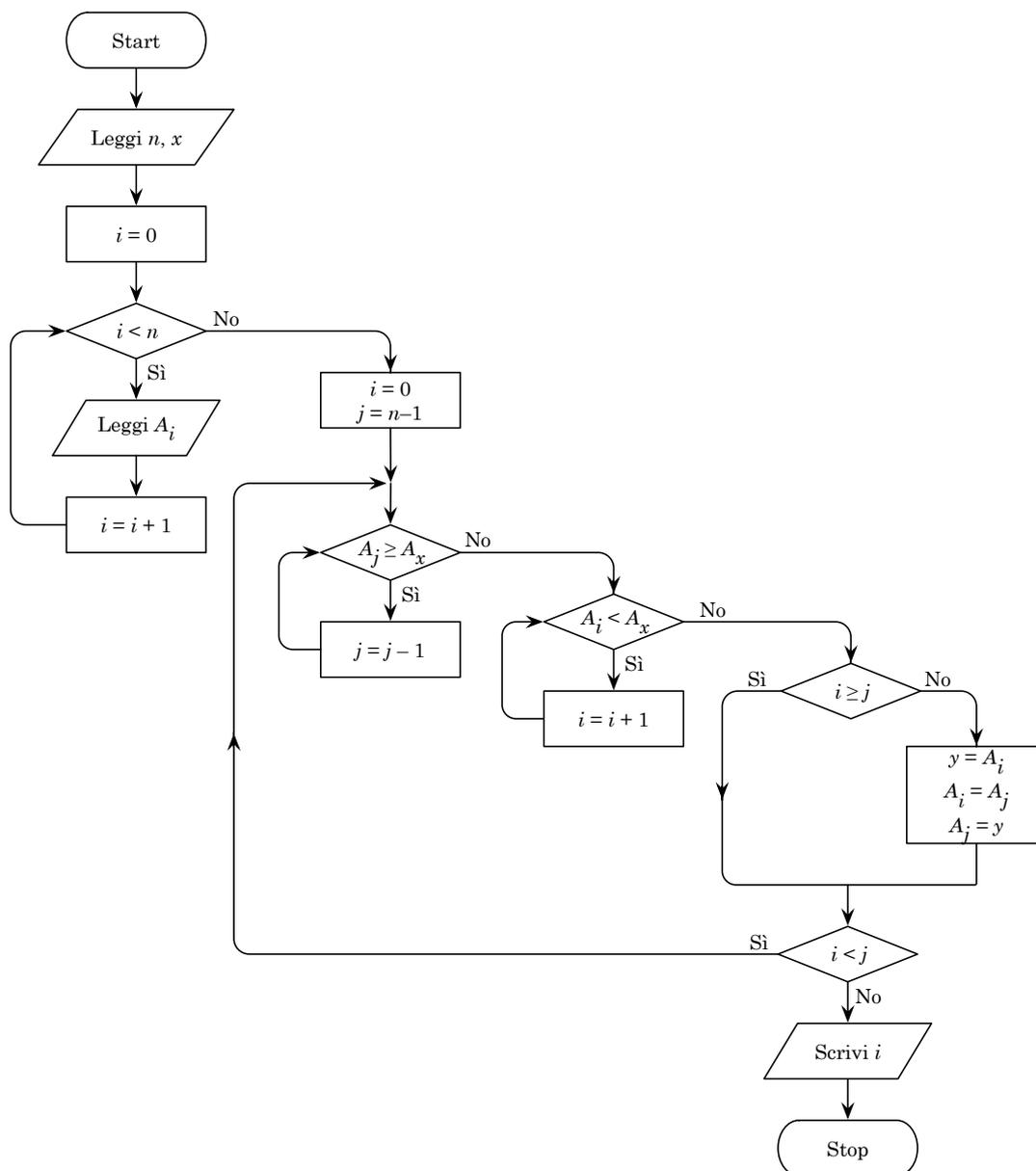
```
1 int main(void) {
2     int n, i, j, k, x, max, A[100], B[100];
3     ...
4     B[0] = 0;
5     for (i=0; i<n; i++) {
6         k = B[0]+1;
7         B[0] = B[0]+1;
8         B[k] = A[i];
9         while (k>1 && B[k/2]<B[k]) {
10            x = B[k];
11            B[k] = B[k/2];
12            B[k/2] = x;
13            k = k/2;
14        }
15    }
16    for (i=n-1; i>=0; i--) {
17        max = B[1];
18        k = B[0];
19        B[1] = B[k];
20        k = k-1;
21        B[0] = B[0]-1;
22        j = 1;
23        while (2*j<=k && (B[j]<B[2*j] || B[j]<B[2*j+1])) {
24            x = B[j];
25            if (B[2*j] > B[2*j+1]) {
26                B[j] = B[2*j];
27                B[2*j] = x;
28                j = 2*j;
29            } else {
30                B[j] = B[2*j+1];
31                B[2*j+1] = x;
32                j = 2*j+1;
33            }
34        }
35        printf("%d ", max);
36    }
37    return(0);
38 }
```

Soluzione



Esercizio n. 4 (max 6 punti)

Codificare in linguaggio C un programma completo che implementi l'algoritmo descritto dal seguente diagramma di flusso. Nella codifica porre attenzione anche alla definizione delle variabili e degli array.



Soluzione

```
1 #include <stdlib.h>
2 #include <stdio.h>
3 #define MAX 100
4
5 int main(void) {
6     int i, j, n, x, y, A[MAX];
7     scanf("%d %d", &n, &x);
8     for (i=0; i<n; i++)
9         scanf("%d", &A[i]);
10    i = 0;
11    j = n-1;
12    do {
13        while (A[j]>=A[x])
14            j--;
15        while (A[i]<A[x])
16            i++;
17        if (i<j) {
18            y = A[i];
19            A[i] = A[j];
20            A[j] = y;
21        }
22    } while (i<j);
23    printf("%d\n", i);
24    return(0);
25 }
```