

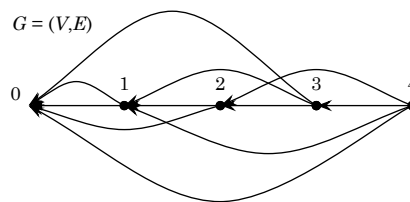
Esame scritto del 27 giugno 2008

Esercizio n. 1

Letto in input un intero $n > 0$ costruire le liste di adiacenza di un grafo orientato $G = (V, E)$, tale che $V = \{0, 1, 2, \dots, n-1\}$ ed $E = \{(i, j) : 0 \leq j < i < n\}$. Stampare le liste di adiacenza del grafo G .

Esempio Sia $n = 5$. Si chiede quindi di costruire e stampare le liste di adiacenza di un grafo orientato come in figura, con insieme dei vertici $V = \{0, 1, 2, 3, 4\}$ e con insieme degli spigoli

$$E = \{(1, 0), (2, 0), (2, 1), (3, 0), (3, 1), (3, 2), (4, 0), (4, 1), (4, 2), (4, 3)\}$$



Soluzione

```

1  #include <stdlib.h>
2  #include <stdio.h>
3  #define MAX 100
4
5  struct nodo {
6      int info;
7      struct nodo *next;
8  };
9
10 void stampaLista(struct nodo *p) {
11     while (p != NULL) {
12         printf("%d --> ", p->info);
13         p = p->next;
14     }
15     printf("NULL\n");
16     return;
17 }
18
19 void stampaGrafo(struct nodo *G[ ], int n) {
20     int i;
21     for (i=0; i<n; i++) {
22         printf("%3d: ", i);
23         stampaLista(G[i]);

```

```

24     }
25     return;
26 }
27
28 void aggiungiSpigoloOrientato(struct nodo *G[ ], int u, int v) {
29     struct nodo *p;
30     p = malloc(sizeof(struct nodo));
31     p->info = v;
32     p->next = G[u];
33     G[u] = p;
34     return;
35 }
36
37 void costruisciGrafo(struct nodo *G[ ], int n) {
38     int i, j;
39     for (i=n-1; i>=0; i--) {
40         G[i] = NULL;
41         for (j=0; j<i; j++)
42             aggiungiSpigoloOrientato(G, i, j);
43     }
44     return;
45 }
46
47 int main(void) {
48     int n;
49     struct nodo *G[MAX];
50     printf("inserisci un intero: ");
51     scanf("%d", &n);
52     costruisciGrafo(G, n);
53     stampaGrafo(G, n);
54     return(0);
55 }

```

Esercizio n. 2

Letti in input due interi n ed m , costruire una matrice A di n righe ed m colonne con numeri interi casuali compresi tra 15 e 72 (estremi inclusi). Stampare la matrice A . Scambiare l'ordine delle colonne della matrice in modo tale che per prime vengano riportate quelle i cui elementi hanno somma pari e dopo quelle con somma dispari. Stampare la matrice A così modificata.

Esempio Sia $n = 3$ e $m = 5$ e si consideri la seguente matrice di numeri casuali:

$$A = \begin{pmatrix} 15 & 60 & 20 & 17 & 64 \\ 22 & 50 & 33 & 53 & 18 \\ 34 & 40 & 70 & 21 & 56 \end{pmatrix}$$

La seconda e la quinta colonna hanno elementi la cui somma è pari, quindi rielaborando la matrice si ottiene, ad esempio, la seguente:

$$A = \begin{pmatrix} 64 & 60 & 20 & 17 & 15 \\ 18 & 50 & 33 & 53 & 22 \\ 56 & 40 & 70 & 21 & 34 \end{pmatrix}$$

Soluzione

```
1 #include <stdlib.h>
2 #include <stdio.h>
3 #include <time.h>
4 #define MAX 30
5
6 void scambia(int *a, int *b) {
7     int c;
8     c = *a;
9     *a = *b;
10    *b = c;
11    return;
12 }
13
14 void generaMatrice(int A[MAX][MAX], int n, int m) {
15     int i, j;
16     srand((unsigned)time(NULL));
17     for (i=0; i<n; i++)
18         for (j=0; j<m; j++)
19             A[i][j] = (rand() % 58) + 15;
20     return;
21 }
22
23 void stampaMatrice(int A[MAX][MAX], int n, int m) {
24     int i, j;
25     for (i=0; i<n; i++) {
26         for (j=0; j<m; j++)
27             printf("%2d ", A[i][j]);
28         printf("\n");
29     }
30     printf("\n");
31     return;
32 }
33
34 int sommaColonna(int A[MAX][MAX], int n, int m, int j) {
35     int i, s=0;
```

```

36     for (i=0; i<n; i++)
37         s = s + A[i][j];
38     return(s);
39 }
40
41 void scambiaColonne(int A[MAX][MAX], int n, int m, int h, int k) {
42     int i;
43     for (i=0; i<n; i++)
44         scambia(&A[i][h], &A[i][k]);
45     return;
46 }
47
48 void spostaColonne(int A[MAX][MAX], int n, int m) {
49     int i=0, j=m-1, x;
50     while (i<j) {
51         x = sommaColonna(A, n, m, i);
52         while (x % 2 == 0 && i<j) {
53             i++;
54             x = sommaColonna(A, n, m, i);
55         }
56         x = sommaColonna(A, n, m, j);
57         while (x % 2 == 1 && i<j) {
58             j--;
59             x = sommaColonna(A, n, m, j);
60         }
61         if (i<j)
62             scambiaColonne(A, n, m, i, j);
63     }
64     return;
65 }
66
67 int main(void) {
68     int A[MAX][MAX], n, m;
69     printf("Numero di righe e di colonne: ");
70     scanf("%d %d", &n, &m);
71     generaMatrice(A, n, m);
72     stampaMatrice(A, n, m);
73     spostaColonne(A, n, m);
74     stampaMatrice(A, n, m);
75     return(0);
76 }

```