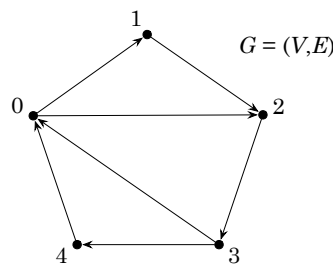


Esame scritto del 7 febbraio 2007

Esercizio n. 1

Letto in input un grafo orientato $G = (V, E)$ rappresentarlo utilizzando delle liste di adiacenza. Letta in input una lista di vertici L , verificare se tale sequenza rappresenta un cammino euleriano su G . Ricordiamo che un cammino su un grafo è euleriano se consente di percorrere una sola volta ogni spigolo del grafo (non è richiesto che il cammino sia semplice).

Esempio Si consideri il grafo G rappresentato in figura. La sequenza $L : 3 \rightarrow 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 0 \rightarrow 2$ rappresenta un cammino euleriano su G , mentre le sequenze $L' : 3 \rightarrow 0 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 1$ e $L'' : 1 \rightarrow 2 \rightarrow 0 \rightarrow 4 \rightarrow 3 \rightarrow 0 \rightarrow 2 \rightarrow 3$ non rappresentano cammini euleriani.



Soluzione

```

#include <stdlib.h>
#include <stdio.h>
#define MAX 50

struct nodo {
    int info, flag;
    struct nodo *next;
};

struct nodo *leggi_lista(void) {
    struct nodo *p, *primo=NULL, *ultimo=NULL;
    int i, n;

    printf("Numero di elementi: ");
    scanf("%d", &n);
    printf("Inserisci %d elementi: ", n);
    for (i=0; i<n; i++) {
        p = malloc(sizeof(struct nodo));
        scanf("%d", &p->info);
        p->next = NULL;
        p->flag = 0;
        if (primo == NULL)
            primo = p;
    }
}

```

```

        if (ultimo != NULL)
            ultimo->next = p;
        ultimo = p;
    }
    return(primo);
}

int leggi_grafo(struct nodo *G[]) {
    int i, n;

    printf("Numero di vertici del grafo: ");
    scanf("%d", &n);
    for (i=0; i<n; i++) {
        printf("Lista dei vertici adiacenti al vertice %d:\n", i);
        G[i] = leggi_lista();
    }
    return(n);
}

int euleriano(struct nodo *G[], int n, struct nodo *L) {
    struct nodo *p, *q;
    int flag = 1, i;

    p = L;
    while (p->next != NULL && flag == 1) {
        q = G[p->info];
        while (q != NULL && q->info != p->next->info)
            q = q->next;
        if (q == NULL || q->flag == 1)
            flag = 0;
        else
            q->flag = 1;
        p = p->next;
    }
    for (i=0; i<n && flag; i++) {
        p = G[i];
        while (p != NULL && p->flag == 1)
            p = p->next;
        if (p != NULL)
            flag = 0;
    }
    return(flag);
}

int main(void) {
    struct nodo *G[MAX], *L;
    int n;

    n = leggi_grafo(G);
    printf("Inserisci il cammino L\n");
    L = leggi_cammino();
    if (euleriano(G, n, L))
        printf("Il cammino L e' euleriano.\n");
    else
        printf("Il cammino L non e' euleriano.\n");
    return(0);
}

```

Esercizio n. 2

Letta in input una stringa s di caratteri, devo digitare quella parola in un messaggio SMS utilizzando un telefono cellulare privo del sistema “T9” con la tastiera riportata in figura. Ad esempio quindi, per visualizzare il carattere “f” dovrò premere per tre volte il tasto “2”. Stampare la sequenza di tasti che devono essere premuti per ottenere la stringa s .

Esempio Si consideri la tastiera di un telefonino riprodotta in figura.

1 abc	2 def	3 ghi
4 jkl	5 mno	6 pqr
7 stu	8 vwx	9 yz

Per stampare la parola $s = \text{“pappagallo”}$ si deve digitare la seguente sequenza di tasti:

$\underbrace{p}_{6} \underbrace{a}_{1} \underbrace{p}_{6} \underbrace{p}_{6} \underbrace{a}_{1} \underbrace{g}_{3} \underbrace{a}_{1} \underbrace{l}_{444} \underbrace{l}_{444} \underbrace{o}_{555}$

Soluzione

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

int main(void) {
    int c[26], i, j;
    char s[20];

    for (i=0; i<26; i = i+3)
        for (j=i; j<i+3; j++)
            c[j] = i/3 + 1;
    printf("Inserisci una parola: ");
    scanf("%s", s);
    printf("Devi digitare i seguenti tasti: ");
    for (i=0; i<strlen(s); i++)
        for (j=0; j<=(s[i]-97)%3; j++)
            printf("%d ", c[s[i]-97]);
    printf("\n");
    return(0);
}
```