

Esercitazione di Laboratorio 30/05/03

1

correggere il programma in modo che possa essere compilato senza errori o warning.

```
#include <stdio.h>
int b;
void funz (int *b, int a)
{b = a + 1;
 return(5)}
int main()
{int a, b;
 for(a=4; a<8; a++) funz(a,b);
 return(0);}
```

2

Data la seguente definizione di tipo:

```
typedef int vett[100];
```

scrivere una funzione C che dato un parametro A di tipo **vett**, restituisce attraverso il valore di ritorno od un opportuno parametro il seguente risultato: 1 se A contiene almeno due elementi uguali, altrimenti 0.

```
#include <stdio.h>
typedef int vett[100];
int trova (vett A, int ret)
{ int i,j;
 for(i=0; (i<99) && (ret==0); i++)
 for(j=0; j<100; j++)
 if (A[i] == A[j])
 ret=1;
 return(ret); }

main()
{vett B;
 int i;
 for(i=0;i<50;i++)
 B[i]=i;
 for(;i<100;i++)
 B[i]=10;
 printf("%d \n",trova(B, &i));
}
```

3

Data la seguente definizione di tipo:

```
typedef int vett[100];
```

scrivere una funzione C che dato un parametro A di tipo **vett** ed un parametro N di tipo **int** restituisce attraverso il valore di ritorno: 100 se A non contiene N, altrimenti il numero di elementi di A che precedono N, cioè che si trovano nelle posizioni di A precedenti la posizione dove si trova N.

```
#include <stdio.h>
typedef int vett[100];
int Trova (vett A, int N)
{ int i,ret = 100;
 for(i=0; i<100; i++)
 if(i == N)
 ret = i;
 return(ret);
}
```

```

main() /*solo per provare*/
{ vett V;
  int i;
  for(i=0;i<100; i++)
    V[i]=i;
  printf("%d\t ",Trova(V, 50));
  printf("%d\n", Trova(V,200));
  return(0);
}

```

4

Il programma che segue vuole ordinare in maniera crescente un vettore rispetto al campo a della seguente struttura:

```
typedef struct {int a,b;} ELE;
```

correggere il codice in modo da rendere possibile ciò. L'algoritmo usa il metodo di selezione per ordinare.

```

#include <stdio.h>
#define DIM 3
typedef struct {int a,b;} ELE;
void carica (ELE vt[])/corretta*/
{int i;
  for(i=0; i<DIM; i++)
    scanf("%d%d",&vt[i].a,&vt[i].b);
}
void stampa (ELE vt[]) /*corretta*/
{int i;
  for(i=0; i<DIM; i++)
    printf("%d\t%d\n",vt[i].a,vt[i].b);}

void ordina (ELE vt[])/errata*/
{int i, j;
  int temp;
  for (i=0; i < DIM-1; i++)
    for (j=0; j < DIM; j++)
      if (vt[i].a > vt[j].a)
        {temp = vt[i].a;
         vt[i].a = vt[j].a;
         vt[j].a = temp;}
}
int main()
{ ELE v[DIM];
  carica(v); ordina(v); stampa(v); return(0);
}

```

5

Data la seguente struttura dati:

```
typedef struct el {int val; struct el *next;}ELE;
```

correggere le funzioni che seguono in modo che compiano l'azione richiesta.

```

/*inserisce in ordine gli elementi nella lista*/
ELE *carica(ELE *t)
{ELE *el, *aux, *prec;

  el = calloc(1,sizeof(ELE));

  printf("dammi un intero: ");
  scanf("%d",&(el->val));
  el->next = NULL;

  prec = aux = t;
}

```

```

if (t == NULL)
    t = el;
else
    {
        while((aux!=NULL) &&\
            (aux->val<= el->val))
            {prec = aux; aux = aux -> next;}
        el->next = aux; prec->next = el;}
return(t);
}

```

/*stampa gli elementi della lista*/

```

void stampa (ELE *t)
{ELE *aux;
  aux = t;

  while(aux != NULL)
    printf("%d\t",aux->val);
  aux = aux -> next;
  printf("\n");
}

```

/*cancella un elemento dalla lista*/

```

ELE *cancella(ELE *t)
{ELE *el, *aux, *prec;
  int del;
  printf("dammi elemento da canc.: ");
  scanf("%d",&del);
  aux = prec = t;
  while ((aux != NULL) &&(aux->val != del))
    { prec = aux; aux = aux -> next;}
  if (aux != NULL)
    prec->next = aux->next;

  return(t);
}

```

```

int main()
{int i;
  ELE *caput;
  caput=NULL;
  for(i=0;i<5;i++)
    caput = carica(caput);
  stampa(caput);
  caput=cancella(caput);
return(0);}

```

6

Siano dati due file di testo f e g denominati 'n1.txt' e 'n2.txt' contenenti due sequenze, ordinate in modo crescente, di numeri interi disposti uno per riga. Costruire un programma in C che scrive gli elementi di f e g sul file di testo h denominato 'risultato.txt', mantenendo h ordinato in modo crescente.

Esempio:	n1.txt	n2.txt	risultato.txt
	5	3	3
	15	7	5
		15	7
		24	15
			15
			24

```
#include<stdio.h>
#include <stdlib.h>
void main() {
    FILE *f, *g, *h;
    int a, b;
    if ((f=fopen("n1.txt","r")) == NULL)
        exit(-1);
    if ((g=fopen("n2.txt","r")) == NULL)
        exit(-2);
    if ((h=fopen("risultato.txt","w")) == NULL)
        exit(-3);
    fscanf(f,"%d",&a); fscanf(g,"%d",&b);
    while((feof(f) == 0) && (feof(g) == 0))
        if (a > b)
            { fprintf(h,"%d\n",b);
              fscanf(g,"%d",&b);
              fscanf(f,"%d",&a);}
        else
            { fprintf(h,"%d\n",a);
              fscanf(f,"%d",&a);
              fscanf(g,"%d",&b); }
    fclose(f); fclose(g); fclose(h);
}
```