

# INFORMATICA - CdL in FISICA - Esercitazione del 14/03/2008

## ESERCIZIO 1

• Sia  $x = -(75)_{10}$

- Rappresentare  $x$ , se possibile, in complemento a 6 con 3 cifre.
- Rappresentare  $x/c - x$  in complemento a 5 su 3 cifre.

## ESERCIZIO 2

Riscrivere il seguente frammento di codice utilizzando solo i seguenti costrutti di controllo: assegnamento, sequenza, blocco, `while` e `if-else`.

```
...
for(x=0; x>w; x=x-1)
  for(y=1; y<=t; y++)
    { scanf("%d", &a);
      j=j+a;
    }
...
```

## ESERCIZIO 3

Scrivere un programma che legge un valore intero  $x$  stampa il valore della funzione  $f(x)$  definita come segue:

$$f(x) = \begin{cases} x^2 + 1 & \text{se } 0 \leq x \leq 10 \wedge \text{divide}(x, 5) \\ x + 1 & \text{se } x > 10 \\ \frac{x+1}{2} & \text{altrimenti} \end{cases}$$

## ESERCIZIO 4

Scrivere un programma che legge una sequenza di caratteri che termina con `\n` e stampa:

- la lunghezza complessiva della sequenza (escluso il `\n` finale)
- il numero di cifre, il numero di lettere e il numero di caratteri non alfanumerici che occorrono nella sequenza
- la posizione nella sequenza della prima cifra (-1 se non ci sono cifre);
- la posizione nella sequenza della prima lettera (-1 se non ci sono lettere);
- la posizione nella sequenza dell'ultimo carattere non alfanumerico (-1 se non ce ne sono).

Se ad esempio la sequenza digitata dall'utente è la seguente

```
;Ab8c dE4%44;6Tr.g-T\n
```

un possibile output del programma è:

```
Lunghezza seq. : 20
Num. cifre: 5
Num. lettere: 9
Num. non alfanumerici: 6
Posizione prima cifra: 4
Posizione prima lettera: 2
Posizione ultimo non alfanumerico: 19
```

### ESERCIZIO 5

Completare il seguente programma C in modo che, al termine della sua esecuzione, gli elementi del vettore `vet` siano distribuiti nel vettore medesimo come segue:

- tutte i valori pari si trovano nella porzione iniziale del vettore;
- tutti i valori dispari si trovano nella porzione finale del vettore.

```
#define DIM ...
```

```
main()
```

```
{
```

```
int vet[DIM];
```

```
int i;
```

```
/* AGGIUNGERE QUI EVENTUALI ALTRE DICHIARAZIONI DI VARIABILE */
```

```
for (i=0; i<DIM; i++)
```

```
scanf("%d", &vet[i]);
```

```
/* CODICE DA COMPLETARE */
```

```
}
```

Se, ad esempio, dopo il ciclo di lettura il vettore `vet` è il seguente

2	15	3	8	5	10	-3	21	4
---	----	---	---	---	----	----	----	---

una possibile soluzione consiste nel redistribuire i valori di `vet` come segue:

2	8	10	4	15	3	5	-3	21
---	---	----	---	----	---	---	----	----

**Nota bene:** non è consentito utilizzare altre variabili strutturate oltre alla variabile `vet`.

# Simulazione Compattiva:

$$1) \quad x = -(75)_{10}$$

$$b=6 \quad n=3 \quad 6^3 = 36 \cdot 3 = \underline{216} \quad [-108, 108)$$

$$75 = 2 \cdot 36 + 0 \cdot 6 + 3 \cdot 1$$

$$i) \quad (75)_6 = 203 \quad \rightarrow \quad -(75)_6 = 353$$

$$75 = 3 \cdot 25 + 0 \quad 5^3 = 125 \quad [-62, 62)$$

ii) non rappresentabile

2) for ( $x=0$ ;  $x > w$ ;  $x = x-1$ )

$y=1$ ;  
while ( $y \leq t$ )

{ scanf (...);

$j = j+a$ ;  
 $y = y+1$ ;

$x=0$

while ( $x > w$ )

{  $y=1$ ;

while ( $y \leq t$ )

{ scanf ("%d", &a);

$j = j+a$ ;

$y = y+1$ ;

{  
 $x = x-1$ ;

}

3) #include <stdio.h>

main() {

int x; float f-x;

printf("inserire il valore di x");

scanf("%d",&x);

if (x > 10)

f-x = x+1;

else if (x >= 0)

f(x = x\*x + 1;

else f-x = (x+1)/2.0;

per fare tutto float.

printf("f(%d) = %f", x, f-x);

4) #include <stdio.h>

main() {

char c;

int lung = 0; int num\_cifre = 0; int num\_lett = 0;

int non\_alfa = 0; pos\_cif = -1; pos\_lett = -1; pos\_nonalfa = -1;

do {

c = getchar();

if (c != '\n')

lung ++;

if (c >= '0' && c <= '9')

controlla le cifre

{ num\_cifre ++;

if (num\_cifre == 1)

pos\_cif = lung;

}

controlla le lettere

else if ((c >= 'a' && c <= 'z') || (c >= 'A' && c <= 'Z'))

{ num\_lett ++;

if (num\_lett == 1)

pos\_lett = lung;

}

else { non\_alfa ++;

pos\_nonalfa = lung; } while (c != '\n')

```

printf("Lunghezza seq. : %d", lung);
printf("Num. cifre : %d", num_cifre);
printf("Num. lettere : %d", num_lett);
printf("Num. non alfanumerici : %d", non_alfa);
printf("
ecc. ecc.
}

```

5) #include <stdio.h>

#define DIM ...

main() {

int vet[DIM];

int i; int appoggio; int primo\_dispari = 0;

for (i=0; i < DIM; i++) ← acquisisce il vettore  
scanf("%d", &vet[i]);

for (i=0; i < DIM; i++)

if (vet[i] % 2 == 0)

if (i == primo\_dispari) ← aggiorna primo dispari

primo\_dispari ++

else { appoggio = vet[primo\_dispari]; ← scambia

vet[primo\_dispari] = vet[i];

vet[i] = appoggio;

primo\_dispari ++;

for (i=0; i < DIM; i++) ← stampa vet ordinata

printf("%d", vet[i]);

}