

Docente: Monica Nesi

1. Scrivere un programma in C che, dati in ingresso due numeri naturali, calcola il loro prodotto utilizzando la somma e stampa il risultato.

```
#include <stdio.h>
main()
{
    int x, y, z;
    scanf("%d%d", &x, &y);
    z = 0;
    while (y > 0)
    {
        z = z+x;
        y = y-1;
    }
    printf("%d\n", z);
}
```

2. Scrivere un programma in C che calcola l'area di un cerchio di raggio  $r$  dato in ingresso.

```
#include <stdio.h>
main()
{
    const float PiGreco = 3.14;
    float r, area;
    scanf("%f", &r);
    area = PiGreco*r*r;
    printf("L'area del cerchio di raggio %f e' pari a %f.\n", r, area);
}
```

3. Scrivere un programma in C che calcola il volume totale di bottiglie e lattine di una data capacità e stampa il risultato.

```
#include <stdio.h>
main()
{
    const float bvolum = 2.0, lvolum = 0.33;
    int b,l;
    float totale;
    scanf("%d%d", &b, &l);
    totale = b*bvolum + l*lvolum;
    printf("Il volume totale e' di %f litri.\n", totale);
}
```

---

<sup>1</sup>Si prega di segnalare ogni eventuale errore all'indirizzo di posta elettronica [monica@di.univaq.it](mailto:monica@di.univaq.it).

4. Scrivere un programma in C che, data in ingresso una sequenza di numeri interi terminata da uno 0, scrive in uscita la lunghezza della sequenza (ovvero quanti sono i numeri della sequenza prima di 0).

```
#include <stdio.h>
main()
{
    int x, cont;
    scanf("%d", &x);
    cont = 0;
    while (x != 0)
    {
        cont = cont+1;
        scanf("%d", &x);
    }
    printf("%d\n", cont);
}
```

5. Scrivere un programma in C che, data in ingresso una sequenza di numeri interi terminata da uno 0, scrive 1 in uscita non appena trova un numero negativo nella sequenza, altrimenti scrive 0.

```
#include <stdio.h>
main()
{
    int x, trovato;
    scanf("%d", &x);
    trovato = 0;
    while (x != 0 && trovato == 0)
    {
        if (x < 0) trovato = 1;
        else scanf("%d", &x);
    }
    printf("%d\n", trovato);
}
```

6. Modificare il programma precedente in modo da stampare in uscita la posizione, all'interno della sequenza, del primo numero negativo nel caso in cui ne esista almeno uno.

```
#include <stdio.h>
main()
{
    int x, trovato, cont;
    scanf("%d", &x);
    trovato = 0;
    cont = 0;
    while (x != 0 && trovato == 0)
    {
        cont = cont+1;
        if (x < 0) trovato = 1;
        else scanf("%d", &x);
    }
    if (trovato == 0)
        printf("Non esistono numeri negativi nella sequenza data in ingresso.\n");
}
```

```

    else
        printf("Il primo numero negativo si trova in posizione n.%.d.\n", cont);
}

```

7. Scrivere un programma in C che, data in ingresso una sequenza di numeri interi terminata da uno 0, stampa il minimo numero della sequenza.

```

#include <stdio.h>
main()
{
    int x, min;
    scanf("%d", &x);
    min = x;
    while (x != 0)
    {
        if (x < min) min = x;
        scanf("%d", &x);
    }
    printf("Il minimo della sequenza e' %.d.\n", min);
}

```

8. Con riferimento alla specifica dell'esercizio precedente, scrivere un programma che calcola il massimo della sequenza.

```

#include <stdio.h>
main()
{
    int x, max;
    scanf("%d", &x);
    max = x;
    while (x != 0)
    {
        if (x > max) max = x;
        scanf("%d", &x);
    }
    printf("Il massimo della sequenza e' %.d.\n", max);
}

```

9. Scrivere un programma in C che, data in ingresso una sequenza di numeri interi terminata da uno 0, stampa in uscita la stessa sequenza data in ingresso con i numeri positivi cambiati di segno. Ad esempio, data in ingresso la sequenza 3 -5 8 -4 7 0, il programma stampa la sequenza -3 -5 -8 -4 -7.

```

#include <stdio.h>
main()
{
    int x;
    scanf("%d", &x);
    while (x != 0)
    {
        if (x > 0 ) printf("%d ", -x);
        else printf("%d ", x);
        scanf("%d", &x);
    }
    printf("\n");
}

```

10. Scrivere un programma in C che, dato in ingresso un numero intero positivo  $n$ , stampa in uscita tutti i divisori di  $n$  (inclusi 1 ed  $n$ ).

```
#include <stdio.h>
main()
{
    int n, cont;
    cont = 1;
    scanf("%d", &n);
    while (cont <= n/2)
    {
        if (n%cont == 0 ) printf("%d ", cont);
        cont = cont+1;
    }
    printf("%d\n", n);
}
```

11. Scrivere un programma in C che, dato in ingresso un numero intero positivo  $n$ , stampa in uscita la stringa "Il numero  $n$  è un numero primo" se  $n$  è un numero primo, altrimenti stampa la stringa "Il numero  $n$  non è un numero primo".

```
#include <stdio.h>
main()
{
    int n, cont, div;
    cont = 2;
    div = 0;
    scanf("%d", &n);
    if (n == 1) printf("Il numero %d non e' un numero primo.\n", n);
    else
    { while (cont <= n/2 && div == 0)
        {
            if (n%cont == 0 ) div = 1;
            else cont = cont+1;
        }
        if (div == 0)
            printf("Il numero %d e' un numero primo.\n", n);
        else
            printf("Il numero %d non e' un numero primo.\n", n);
    }
}
```

12. Scrivere un programma in C che, dati in ingresso due numeri interi positivi, calcola e stampa il loro massimo comun divisore (vedi Esempio 3.7 del testo di riferimento).

*Algoritmo1:*

```
#include <stdio.h>
main()
{
    int n, m, mcd, min, cont;
    scanf("%d%d", &n, &m);
    mcd = 1;
    if (n <= m) min = n;
    else min = m;
```

```

    cont = 1;
    while (cont <= min)
    {
        if (n%cont == 0 && m%cont == 0) mcd = cont;
        cont = cont+1;
    }
    printf("MCD(%d,%d) = %d.\n", n, m, mcd);
}

```

*Algoritmo2:*

```

#include <stdio.h>
main()
{
    int n, m, x, y, mcd, cont;
    scanf("%d%d", &n, &m);
    x = n;
    y = m;
    while (y != x)
    {
        if (y > x) y = y-x;
        else x = x-y;
    }
    mcd = x;
    printf("MCD(%d,%d) = %d.\n", n, m, mcd);
}

```

13. Scrivere un programma in C che, dati in ingresso due numeri interi positivi, calcola e stampa il loro minimo comune multiplo (Esercizio 3.6 del testo di riferimento).

```

#include <stdio.h>
main()
{
    int n, m, cont, trovato, prodotto;
    scanf("%d%d", &n, &m);
    trovato = 0;
    prodotto = n*m;
    if (n > m) cont = n;
    else cont = m;
    while (cont < prodotto && trovato == 0)
    {
        if (cont%n == 0 && cont%m == 0) trovato = 1;
        else cont = cont+1;
    }
    printf("mcm(%d,%d) = %d.\n", n, m, cont);
}

```

14. Scrivere un programma in C che, data in ingresso una sequenza di numeri interi terminata da uno 0, il cui primo numero è diverso da 0, stampa in uscita 1 se il primo numero occorre nel resto della sequenza, altrimenti stampa 0. Ad esempio, data in ingresso la sequenza 3 -5 4 3 -10 0, il programma stampa 1 mentre, data la sequenza 2 -5 4 3 -10 0, il programma stampa 0.

```

#include <stdio.h>
main()
{
    int x, y, trovato;
    scanf("%d%d", &x, &y);
    trovato = 0;
    while (y != 0 && trovato == 0)
    {
        if (x == y) trovato = 1;
        else scanf("%d", &y);
    }
    printf("%d\n", trovato);
}

```

*Versione con array* (Hp. la lunghezza massima della sequenza in ingresso è 20)

```

#include <stdio.h>
#define lung 20
main()
{
    int i, j, trovato;
    int a[lung];
    i = 0;
    scanf("%d", &a[i]);
    while (a[i] != 0 && i < lung)
    {
        i = i+1;
        scanf("%d", &a[i]);
    }
    j = 1;
    trovato = 0;
    while (j < i && trovato == 0)
    {
        if (a[0] == a[j]) trovato = 1;
        else j = j+1;
    }
    printf("%d\n", trovato);
}

```

15. Con riferimento alla specifica dell'esercizio precedente, scrivere un programma che stampa in uscita quante volte il primo numero occorre nel resto della sequenza.

```

#include <stdio.h>
main()
{
    int x, y, cont;
    scanf("%d%d", &x, &y);
    cont = 0;
    while (y != 0)
    {
        if (x == y) cont = cont+1;
        scanf("%d", &y);
    }
    printf("Il numero %d occorre %d volta/e nel resto della sequenza.\n",

```

```

        x, cont);
    }

```

16. Scrivere un programma in C che, data in ingresso una sequenza (non vuota) di numeri interi terminata da uno 0, stampa in uscita 1 se tutti i numeri sono pari, altrimenti stampa 0. Ad esempio, data la sequenza 4 -10 0, il programma stampa 1 mentre, data la sequenza -6 10 3 2 0, il programma stampa 0. **N.B.** Il valore in uscita 0 o 1 viene stampato una volta sola.

```

#include <stdio.h>
main()
{
    int x, pari;
    scanf("%d", &x);
    pari = 1;
    while (x != 0 && pari == 1)
    {
        if (x%2 != 0) pari = 0;
        else scanf("%d", &x);
    }
    printf("%d\n", pari);
}

```

17. Scrivere un programma in C che, data in ingresso una sequenza di numeri interi, stampa in uscita in ordine invertito i primi 10 numeri della sequenza.

```

#include <stdio.h>
#define lung 10
main()
{
    int i;
    int a[lung];
    i = 0;
    while (i < lung)
    {
        scanf("%d", &a[i]);
        i = i+1;
    }
    while (i > 0)
    {
        printf("%d ", a[i-1]);
        i = i-1;
    }
    printf("\n");
}

```

18. Scrivere un programma in C che, dati in ingresso due numeri interi positivi  $m$  ed  $n$  (con  $m < n$ ), stampa la sequenza di tutti i numeri pari compresi tra  $m$  ed  $n$  ( $m$  ed  $n$  inclusi, se pari). Ad esempio, se  $m = 5$  ed  $n = 14$  viene stampata la sequenza 6 8 10 12 14.

```

#include <stdio.h>
main()
{
    int m, n, x;

```

```

scanf("%d%d", &m, &n);
if (m%2 == 0) x = m;
else x = m+1;
while (x <= n)
{
    printf("%d ", x);
    x = x+2;
}
printf("\n");
}

```

## Codice in linguaggio macchina

### *Esercizio 1*

Sia dato il seguente frammento di programma in linguaggio macchina, memorizzato a partire dalla locazione di memoria di indirizzo 150:

```

150 LOADA 21
151 LOADB 20
152 DIF
153 STOREA 21
154 LOADA 20
155 MUL
156 STOREA 20

```

Supponendo che nelle locazioni di indirizzo 20 e 21 siano presenti i valori interi generici  $a$  e  $b$  rispettivamente, dire qual è il contenuto delle locazioni 20 e 21 dopo l'esecuzione del frammento di programma.

### *Soluzione*

Dopo l'esecuzione del frammento di programma, la cella di indirizzo 20 contiene il valore di  $a^2$ , mentre la cella di indirizzo 21 contiene il valore di  $b-a$ .

### *Esercizio 2*

Sia dato il seguente frammento di programma in linguaggio macchina, memorizzato a partire dalla locazione di memoria di indirizzo 200:

```

200 READ 30
201 READ 31
202 LOADA 30
203 LOADB 31
204 ADD
205 STOREA 31
206 LOADB 31
207 MUL
208 STOREA 30

```

Dati due interi generici  $a$  e  $b$  in ingresso, dire qual è il contenuto delle locazioni 30 e 31 dopo l'esecuzione del programma.

### *Soluzione*

Dopo l'esecuzione del programma, la cella di indirizzo 30 contiene il valore di  $(a+b)^2$ , mentre la cella di indirizzo 31 contiene il valore di  $a+b$ .



### *Esercizio 3*

Sia dato il seguente frammento di programma in linguaggio macchina, memorizzato a partire dalla locazione di memoria di indirizzo 100:

```
100 READ 20
101 READ 21
102 LOADA 20
103 LOADB 20
104 MUL
105 STOREA 20
106 LOADA 21
107 LOADB 21
108 MUL
109 STOREA 21
110 LOADA 20
111 LOADB 21
112 MUL
113 STOREA 20
114 WRITE 20
115 HALT
```

Dati due interi generici  $a$  e  $b$  in ingresso, dire qual è l'espressione calcolata dal programma.

#### *Soluzione*

L'espressione calcolata dal programma è  $a^2 b^2$ .

### *Esercizio 4*

Sia dato il seguente frammento di programma in linguaggio macchina, memorizzato a partire dalla locazione di memoria di indirizzo 300:

```
300 READ 100
301 READ 101
302 READ 102
303 LOADA 101
304 LOADB 102
305 ADD
306 STOREA 101
307 LOADA 100
308 LOADB 100
309 MUL
310 LOADB 101
311 MUL
312 STOREA 100
313 WRITE 100
```

Dati gli interi generici  $a$ ,  $b$  e  $c$  in ingresso, dire qual è l'espressione calcolata dal programma.

#### *Soluzione*

L'espressione calcolata dal programma è  $a^2(b+c)$ .

### *Esercizio 5*

Scrivere un frammento di programma in linguaggio macchina che, dati due interi generici  $a$  e  $b$  in ingresso, stampa in uscita il valore dell'espressione  $(b-a) a$ .

#### *Soluzione*

Supponendo che il frammento di programma sia memorizzato a partire dalla locazione di memoria di indirizzo 300 e che vengano utilizzate le locazioni di memoria di indirizzo 50 e 51, il codice in linguaggio macchina è il seguente:

```
300 READ 50
301 READ 51
302 LOADA 51
303 LOADB 50
304 DIF
305 STOREA 51
306 MUL
307 STOREA 50
308 WRITE 50
```

### *Esercizio 6*

Si supponga che le locazioni di memoria di indirizzo 30 e 31 contengano i valori 0 ed 1 rispettivamente. Sia dato il seguente programma in linguaggio macchina, memorizzato a partire dalla locazione di memoria di indirizzo 20:

```
20 READ 32
21 READ 33
22 LOADA 32
23 LOADB 33
24 DIF
25 JUMPZ 28
26 WRITE 31
27 JUMP 29
28 WRITE 30
29 HALT
```

- i) Dati due interi generici  $a$  e  $b$  in ingresso, dire che cosa fa il programma e quale valore viene scritto in uscita.
- ii) Siano i valori in ingresso 5,5 e 7,5. Qual è il valore del registro A alla fine dell'esecuzione del programma nei due casi? Quale è il valore contenuto nella locazione di indirizzo 32, nei due casi?

### *Soluzione*

- i) Il programma testa se due numeri sono diversi, ovvero scrive 1 se  $a \neq b$ , e 0 se  $a = b$ .
- ii) Il valore contenuto nel registro A dopo l'esecuzione del programma è 0 (primo caso) e 2 (secondo caso). Il valore contenuto nella locazione di indirizzo 32 è 5 e 7 rispettivamente.

### *Esercizio 7*

Sia dato il seguente frammento di programma in linguaggio macchina, memorizzato a partire dalla locazione di memoria di indirizzo 10:

```
10 READ 20
11 READ 21
12 LOADA 20
13 LOADB 21
14 DIF
15 JUMPZ 17
16 JUMP 14
17 STOREA 20
18 WRITE 20
```

Dati due interi generici  $a$  e  $b$  in ingresso, dire che cosa fa il programma e quale valore viene scritto in uscita.

### *Soluzione*

Il programma stampa in uscita 0 se  $a$  è divisibile per  $b$ , altrimenti cicla senza stampare nulla.