



POLITECNICO
MILANO 1863

INFORMATICA Prof. Cassano

A.A. 2017-18

Laboratorio n°1-Soluzioni

Ing. Gian Enrico Conti

Sadegh Astaneh, Ph.D.

Scrivere un programma che, dati due numeri interi positivi inseriti dall'utente, stampi a video i divisori comuni maggiori di 1.

Esempio:

Inserisci due numeri positivi: 18 72

I divisori comuni sono: 2 3 6 9 18

Se l'unico divisore comune sia "1", stampare a video che i due numeri sono coprimi.

Suggerimenti:

usare l'operator % per calcolare il resto della divisione intera

```
//Es1
```

```
int main(int argc, const char * argv[]) {  
  
    int n, m;  
    printf("Inserisci due numeri positivi:\n");  
    scanf("%d", &n);  
    scanf("%d", &m);  
  
    printf("\nI divisori comuni sono: ");  
    int i;  
    int max = n;  
    if (m>n)  
        max= m;  
    for (i=1; i<max; i++)  
    {  
        int restom = m % i;  
        int reston = n % i;  
        if (restom == 0 && reston == 0){  
            printf("%d ", i);  
        }  
    }  
  
    // printf e' bufferizzata, un \n "aiuta":  
    printf("\n");  
  
    return 0;  
}
```

Scrivere un programma che, dato un numero positivo n (definito all' inizio del programma o inserito da tastiera) scriva a video "numero primo" se n e' primo (con "primo si intende che non ha divisori interi a parte se' ed 1)

Hints:

Utilizzare un ciclo all'indietro

```
//Es2
```

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

```
int main(int argc, const char * argv[]) {
```

```
    int i, n;
```

```
    printf("Inserisci un numero positivo:\n");
```

```
    scanf("%d", &n);
```

```
    for (i=n-1; i>0; i--) // inutile testare max..
```

```
    {
```

```
        int reston = n % i;
```

```
        if (reston == 0){
```

```
            break;
```

```
        }
```

```
    }
```

```
    // se qui siamo usciti o a fine ciclo (n=1) o per il break:
```

```
    if (i==1){
```

```
        printf("numero primo\n");
```

```
    }
```

```
    return 0;
```

```
}
```

si scriva in programma che, dato un numero intero positivo, stampi a video i bit che lo rappresentano nel consueto formato della conversione manuale:

es:

n = 12

12|

6 | 0

3 | 0

1 | 1

0 | 1

```
//Es3
```

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

```
int main(int argc, const char * argv[]) {
```

```
    int i, n;
```

```
    printf("Inserisci un numero positivo:\n");
```

```
    scanf("%d", &n);
```

```
    printf("%2d|\n", n);
```

```
    printf("-----\n");
```

```
    for (;n>0;)
```

```
    {
```

```
        int divn = n / 2;
```

```
        int reston = n % 2;
```

```
        printf("%2d|%2d\n", divn, reston);
```

```
        n=divn;
```

```
    }
```

```
    return 0;
```

```
}
```

Scrivere un programma che, dati in ingresso due numeri interi positivi N e K , stampa potenze di N con esponenti da 1 a K .

Esempio:

Base: 4

Esponente massimo: 4

Potenze: 4 16 64 256


```
//Es4
```

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

```
int main(int argc, const char * argv[]) {
```

```
    int i, K, N;
```

```
    printf("Inserisci due numeri positivi:\n");
```

```
    scanf("%d", &N);
```

```
    scanf("%d", &K);
```

```
    printf("\nLe potenze di %d sono: \n", N);
```

```
    long pot = N;
```

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

```
    {
```

```
        printf("%ld ", pot);
```

```
        pot*=N;
```

```
    }
```

```
    printf("\n");
```

```
    return 0;
```

```
}
```

Nelle gare di tuffi a 5 giudici il punteggio finale è doppio della somma dei voti ottenuta eliminando il più alto ed il più basso.

Scrivere un programma che legga in input 5 valori e calcoli in uscita il voto finale.

Esempio:

8.0, 7.5, 7.5, 7.5, 7.0 = $22.5 \times 2.0 = 45.0$

(http://www.federnuoto.it/pdf/t_reg_tec_09-13.pdf)

Hints:

Utilizzare un ciclo

Controllare ad ogni iterazione se il voto supera o è inferiore rispettivamente al max o min trovato finora.

```
//ES4
```

```
#include <stdio.h>
int main(int argc, const char * argv[]) {

    int i=0;
    float n, max, min, somma=0;
    for (i=0; i<5; i++)
    {
        printf("Inserisci il %d voto:\n", i+1);
        scanf("%f", &n);

        if (i==0){
            max = n;
            min = n;
        }
        //sommo incondizionato, no so ancora min/max:
        somma+=n;

        if (n>max)
            max=n;
        if (n<min)
            min=n;
    }
    // "scalo" min e max:
    somma= somma - min - max;
    float voto = somma *2;
    printf("%f\n", voto);

    return 0;
}
```

Scrivere un programma che riceve dall'utente un numero N compreso tra 2 e 12 e stampa la tabellina di N nel modo seguente.

Esempio:

Tabellina: 4

$4 \times 1 = 4$

$4 \times 2 = 8$

[...]

$4 \times 10 = 40$

Estensione:

Il programma riceve due interi N e K e stampa tutte le tabelline dei numeri da 1 a N , limitandosi ai primi K valori.

```
//Es6
```

```
#include <stdio.h>  
#include <stdbool.h>
```

```
int main(int argc, const char * argv[]) {
```

```
    int i,n=0;
```

```
    bool ok = false;
```

```
    while (!ok){
```

```
        printf("Inserisci un numero positivo fra 2 e 12:\n");
```

```
        scanf("%d", &n);
```

```
        ok = (n>=2 && n<=12);
```

```
        if (ok)
```

```
            break;
```

```
        else
```

```
            printf("NON e' fra 2 e 12!\n");
```

```
    }
```

```
    printf("\n Tabellina: %d\n", n);
```

```
    for (i=1; i<=10; i++)
```

```
    {
```

```
        printf("%dx%d=%d\n", n, i, n*i);
```

```
    }
```

```
    return 0;
```

```
}
```

si scriva un programma che legga da tastiera i prezzi di 6 articoli espressi in euro.

il programma alla fine deve stampare il prodotto piu caro, il meno caro ed i rispettivi indici di inserimento

es

12

22

34

9

11

23

piu caro: 34, inserito x 3'

meno caro: 9, inserito x 4'

```
//Es7
```

```
#include <stdio.h>
int main(int argc, const char * argv[]) {
    int i=0;
    int indmax, indmin;
    float prezzo, max, min;
    for (i=0; i<6; i++)
    {
        printf("Inserisci il %d prezzo:\n", i+1);
        scanf("%f", &prezzo);

        if (i==0){
            max = prezzo;
            min = prezzo;
            indmax=0;
            indmin=0;
        }
        if (prezzo>max){
            max=prezzo;
            indmax=i;
        }
        if (prezzo<min){
            min=prezzo;
            indmin=i;
        }
    }

    printf("piu caro: %f, inserito x %d\n", max, indmax+1);
    printf("meno caro: %f, inserito x %d\n", min, indmin+1);
    return 0;
}
```

Scrivere un programma che legge un numero intero e stampa video le prima n righe

del triangolo di Floyd (i.e, un triangolo rettangolo che contiene tutti i numeri naturali disposti come segue).Esempio per n=10:

```
1
2 3
4 5 6
7 8 9 10
```

Hint:la soluzione più immediata utilizza due cicli annidati.

Variante:Si progetti anche una soluzione che utilizza un solo ciclo.


```
//Es8
#include <stdio.h>
```

```
int main(int argc, const char * argv[]) {
```

```
    int i,row=1, n, printed=0;
    printf("Inserisci un numero positivo:\n");
    scanf("%d", &n);
    for (i=1; i<=n; i++){
        printf("%d ", i);
        printed++;
        if (printed>=row){
            printf("\n");
            printed=0;
            row++;
        }
    }
}
```

dato in ingresso in numero $n > 0$, il programma stampi un triangolo di asterischi che cominci con 1 asterisco, poi 2 e così via fino a n asterischi

es: $n = 4$

*

**

```
//Es9
```

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

```
int main(int argc, const char * argv[]) {
```

```
    int i=0, n, row = 0, printed = 0;
```

```
    printf("Inserisci un numero positivo:\n");
```

```
    scanf("%d", &n);
```

```
    for (i=0; i<=n; i++){
```

```
        printf("* ");
```

```
        printed++;
```

```
        if (printed>=row){
```

```
            printf("\n");
```

```
            printed=0;
```

```
            row++;
```

```
        }
```

```
    }
```

```
}
```

si scriva un programma che legga in input i cateti di un numero imprecisato di triangoli rettangoli.

L'inserimento termina quando l'utente inserisce cateti per cui l'area del triangolo è ZERO.

A fine elaborazione il programma deve stampare la somma di tutte le aree dei vari triangoli e il valor medio dei perimetri.

Hint: si usi TDP x calcolare la ipotenusa.

```
//Es10
```

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

```
#include <math.h> //to use sqrt.
```

```
int main(int argc, const char * argv[]) {
```

```
    double sommaArea = 0;
```

```
    double sommaPerim = 0;
```

```
    int i;
```

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

```
    {
```

```
        int c1, c2;
```

```
        printf("Inserisci i due cateti:\n");
```

```
        scanf("%d", &c1);
```

```
        scanf("%d", &c2);
```

```
        double area = c1 * c2 / 2;
```

```
        if (area == 0)
```

```
            break;
```

```
        sommaArea += area;
```

```
        // calcolo anche il perimetro, per far cio ipotenusia:
```

```
        double hyp = sqrt (c1*c1 + c2*c2);
```

```
        double perim = c1 + c2 + hyp;
```

```
        sommaPerim+=perim;
```

```
    }
```

```
    double mediaPerim = sommaPerim / i;
```

```
    printf(" %f %f\n", sommaArea, mediaPerim);
```

```
    return 0;
```

```
}
```