

# Fondamenti di Informatica ◇ 2021-22

Laboratorio #1 del 12-10-2021

## Va? Cosa non va?

Scrivere un programma che acquisita una sequenza di valori interi, terminata quando l'utente inserisce il valore 20, calcola e visualizza valor minimo, massimo e media dei valori.

Qualcuno ha proposto la soluzione qui di seguito riportata: verificare che sia una soluzione corretta e ben fatta, modificandola ove necessario nel caso non lo sia (copiate-incollate il codice ...).

```
int main(int argc, char * argv[])
{
    int min, max;
    int val, cont, tot;
    float avg;

    scanf("%d", &val);
    val = min;
    val = max;
    cont = 1;
    scanf("%d", &val);
    while(val != 20){
        if(val < min)
            min = val;
        elseif(val > max)
            max = val;
        tot = tot + val;
        cont++
        scanf("%d", &val);
    }
    avg = tot / cont;
    printf("%d %d %f\n", min, max, avg)

    return 0;
}
```

## Fattori

Scrivere un programma che acquisisce due numeri interi relativi e visualizza 1 se uno è un divisore dell'altro o viceversa, 0 altrimenti. Dopo il valore visualizzato, mettere un 'a-capo'.

### Ingresso/Uscita:

**input:** due numeri interi

**output:** un intero (seguito da un carattere 'a-capo')

### Alcuni casi di test per il collaudo:

**input:** 5542 18

**output:** 0

**input:** 5542 17

**output:** 1

**input:** 13 1950

**output:** 1

## Padding

Si vuole rappresentare a video un valore naturale `num` utilizzando un numero a scelta di cifre `k` inserendo 0 nelle posizioni più significative, fino a raggiungere la dimensione desiderata. Per esempio, volendo rappresentare 842 su 5 cifre, si ottiene 00842.

Scrivere un programma che acquisisce due valori interi entrambi strettamente positivi (e finché non è così richiede il valore che non rispetta il vincolo) `num` e `k`, quindi rappresenta `num` su `k` cifre. Se `k` è minore del numero di cifre presenti in `num`, il programma visualizza il valore `num` come è. Dopo il valore visualizzato, mettere un 'a-capo'.

### Ingresso/Uscita:

**input:** due numeri interi (da verificare)

**output:** un intero (seguito da un carattere 'a-capo')

### Alcuni casi di test per il collaudo:

**input:** 11304 9

**output:** 000011304

**input:** -4 9000 -5 -2 2

**output:** 9000

**input:** 1 1

**output:** 1

---

## Super Mario

Nella preistoria dei videogiochi in Super Mario della Nintendo, Mario deve saltare da una piramide di blocchi a quella adiacente. Proviamo a ricreare le stesse piramidi in C, in testo, utilizzando il carattere cancelletto (`#`) come blocco, come riportato di seguito. In realtà il carattere `#` è più alto che largo, quindi le piramidi saranno un po' più alte.

```
  #  #
 ## ##
### ###
#### ####
```



Notate che lo spazio tra le due piramidi è sempre costituito da 2 spazi, indipendentemente dall'altezza delle piramidi. Inoltre, alla fine delle piramidi **non ci devono essere spazi**. L'utente inserisce l'altezza delle piramidi, che deve essere un valore strettamente positivo e non superiore a 16. In caso l'utente inserisca un valore che non rispetta questi vincoli, la richiesta viene ripetuta.

### Ingresso/Uscita:

**input:** un numero intero (da verificare)

**output:** una sequenza di caratteri

---

## Troncabile primo a destra

Scrivere un programma che acquisisce un valore intero strettamente positivo, e finché non è tale lo richiede. Il programma analizza il valore intero e visualizza 1 nel caso sia un `troncabile primo a destra`, 0 altrimenti. Un numero si dice troncabile primo a destra se il numero stesso e tutti i numeri che si ottengono eliminando una alla volta la cifra meno significativa del numero analizzato al passo precedente, sono numeri primi. Per esempio, se il numero iniziale è 719, i numeri che si ottengono “eliminando una alla volta la cifra meno significativa del numero analizzato al passo precedente ..” sono 71 e 7. Dopo il valore visualizzato, mettere un ‘a-capo’.

### Ingresso/Uscita:

**input:** un intero (da verificare)

**output:** un intero (seguito da un carattere ‘a-capo’)

### Alcuni casi di test per il collaudo:

**input:** 719

**output:** 1

**input:** 473

**output:** 0

**input:** -42 -18 311111

**output:** 0

**input:** 3137

**output:** 1

---