

Esercizio

Sfruttando il costrutto `while` si chieda all'utente di inserire un intero positivo. Il programma terminerà all'inserimento del primo numero negativo.

```
inserire un numero positivo:  
1  
inserire un numero positivo:  
14  
inserire un numero positivo:  
-1
```

Esercizio

```
numero = 0
while numero >= 0 :
    numero = input("inserire un numero positivo: ")
```

WHILE: contatori

É possibile utilizzare una variabile per **contare il numero di iterazioni** svolte:

```
stringa = 'si'
contatore = 0

while stringa == 'si':
    stringa = raw_input("vuoi continuare l'iterazione? si/no")
    contatore = contatore + 1
print 'Hai eseguito ', contatore, ' iterazioni'
```

La variabile che 'conta' il numero di iterazioni si dice **contatore**

variabili di accumulo: media

Spesso si utilizzano i cicli per “accumulare” quantità variabili in una variabile detta d’accumulo o accumulatore.

```
numIterazioni = 5
media = 0
contatore = 0
while contatore < numIterazioni:
    valore = float(input('inserisci un numero '))
    media += valore
    contatore += 1

media /= numIterazioni
print 'il valor medio è ', media
```

WHILE: valore sentinella

Sono valori dello stesso tipo dell'input richiesto all'utente ma sicuramente non parte dell'input atteso che vengono usati per interrompere il ciclo.

```
media = 0
valore = 0
iterazione = -1
while valore >= 0:
    media += valore
    iterazione += 1
    valore = float(input('inserisci un numero positivo (<0 termina
    esecuzione)'))

media /= iterazione
print 'il valor medio è ', media
```

Esercizio

Si costruiscano almeno due programmi che stampino a video i primi 1000 numeri interi dispari.

1, 3, 5, ..., 2001

Esercizio

```
CARDINALITA = 10

numero = 1
while(numero <= 2 * CARDINALITA):
    print "numero = ", numero
    numero = numero + 2
```

```
CARDINALITA = 10

numero = 1
quanti = 0
while(quanti < CARDINALITA):
    if numero % 2 != 0:
        print "numero = ", numero, "quanti = ", quanti
        quanti += 1
    numero = numero + 1
```

Esercizio

```
CARDINALITA = 10

numero = 0
while(numero < CARDINALITA):
    print "numero = ", numero * 2 + 1
    numero = numero + 1
```


Esercizio - varianza

Scrivere un algoritmo per calcolare media e varianza di 10 valori inseriti da tastiera dall'utente.

Esercizio - varianza

Scrivere un algoritmo per calcolare media e varianza di 10 valori inseriti da tastiera dall'utente.

- $$\mu = \frac{\sum_{i=1}^N x_i}{N}$$
- $$\sigma^2 = \frac{\sum_{i=1}^N (x_i - \mu)^2}{N}$$

Esercizio - varianza

Scrivere un algoritmo per calcolare media e varianza di 10 valori inseriti da tastiera dall'utente.

- $\mu = \frac{\sum_{i=1}^N x_i}{N}$
- $\sigma^2 = \frac{\sum_{i=1}^N (x_i - \mu)^2}{N}$
- $\sigma^2 = \frac{\sum_{i=1}^N x_i^2 - \frac{(\sum_{i=1}^N x_i)^2}{N}}{N - 1}$

Esercizio

```
N = 10

somma = 0.0
somma_quadrati = 0.0

contatore = 0
while contatore < N:
    numero = input('inserisci un numero ')
    print "iterazione ", contatore
    somma += float(numero)
    somma_quadrati += float(numero)**2
    contatore += 1

media = somma / N
varianza = (somma_quadrati - (somma**2 / N)) / (N - 1.0)

print "media = ", media
print "varianza = ", varianza
```

Esercizio

Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Esercizio

Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Un numero naturale, maggiore di 1, **si dice primo** se è divisibile solo per se stesso e per l'unità.

Esercizio

Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Un numero naturale, maggiore di 1, **si dice primo** se è divisibile solo per se stesso e per l'unità.

Come posso verificarlo?

Esercizio

Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Un numero naturale, maggiore di 1, **si dice primo** se è divisibile solo per se stesso e per l'unità.

Come posso verificarlo?

- un numero n è divisibile per un divisore d se $n \bmod d = 0$ o equivalentemente $resto = 0$

Esercizio

Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Un numero naturale, maggiore di 1, **si dice primo** se è divisibile solo per se stesso e per l'unità.

Come posso verificarlo?

- un numero n è divisibile per un divisore d se $n \bmod d = 0$ o equivalentemente $\text{resto} = 0$
- ha come unici divisori 1 e se stesso o equivalentemente l'insieme dei divisori $\mathcal{D} \subseteq \mathbb{N}$ deve essere $\mathcal{D} = \{1, n\}$

Esercizio

Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Un numero naturale, maggiore di 1, **si dice primo** se è divisibile solo per se stesso e per l'unità.

Come posso verificarlo?

- un numero n è divisibile per un divisore d se $n \bmod d = 0$ o equivalentemente $\text{resto} = 0$
- ha come unici divisori 1 e se stesso o equivalentemente l'insieme dei divisori $\mathcal{D} \subseteq \mathbb{N}$ deve essere $\mathcal{D} = \{1, n\}$
- posso generare tutti i numeri $d \in \mathbb{N} \mid d \leq n$ e verificare per ciascuno se d è divisore

Esercizio

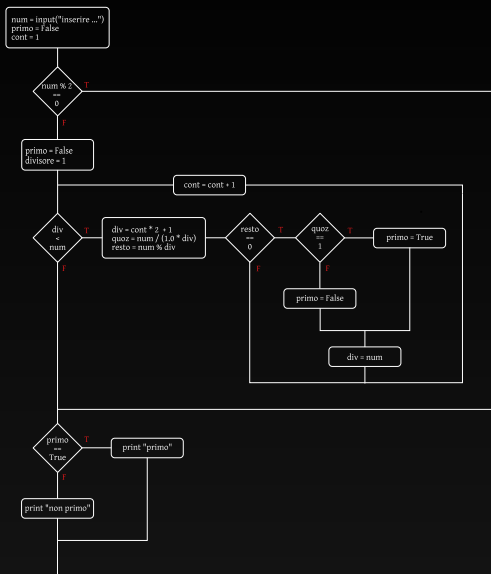
Si costruisca un programma che ricevuto un numero intero riconosca se è primo.

Un numero naturale, maggiore di 1, **si dice primo** se è divisibile solo per se stesso e per l'unità.

Come posso verificarlo?

- un numero n è divisibile per un divisore d se $n \bmod d = 0$ o equivalentemente $\text{resto} = 0$
- ha come unici divisori 1 e se stesso o equivalentemente l'insieme dei divisori $\mathcal{D} \subseteq \mathbb{N}$ deve essere $\mathcal{D} = \{1, n\}$
- posso generare tutti i numeri $d \in \mathbb{N} \mid d \leq n$ e verificare per ciascuno se d è divisore
- al primo divisore diverso da n che incontro, n non è primo.
Devo verificare che $\frac{n}{d} = 1$

Esercizio



Esercizio

```
numero = input("inserire un numero intero positivo: ")
primo = False
contatore = 1

if numero%2 == 0:
    primo = False
    divisore = numero
    print numero, "e' un numero pari"
else:
    divisore = 1.0
    while divisore < numero:
        divisore = contatore * 2 + 1
        quoziente = numero / (1.0 * divisore)
        resto = numero % divisore
        print "numero = ", numero, "contatore = ", contatore
        print "divisore = ", divisore, "quoziente = ", quoziente, "resto = ", resto, "\n"
        if resto == 0:
            if quoziente == 1.0:
                primo = True
                divisore = numero
            else:
                primo = False
                divisore = numero
        contatore = contatore + 1
if primo:
    print numero, "e' un numero primo"
else:
    print numero, " non e' un numero primo"
```