

**Titolo unità didattica:** Concetto di Variabile

[02]

**Titolo modulo :** Operazione di lettura/visualizzazione

[03-T]

Operazioni di base sulle variabili in programmazione

Argomenti trattati:

- ✓ operazione di lettura del valore di una variabile da dispositivo esterno
- ✓ operazione di visualizzazione del valore di una variabile su dispositivo esterno

Prerequisiti richiesti: AP-02-01-T, AP-02-02-T

l'operazione che consente di **associare un valore a una variabile** è l'operazione fondamentale in un algoritmo

**due diverse istruzioni:**

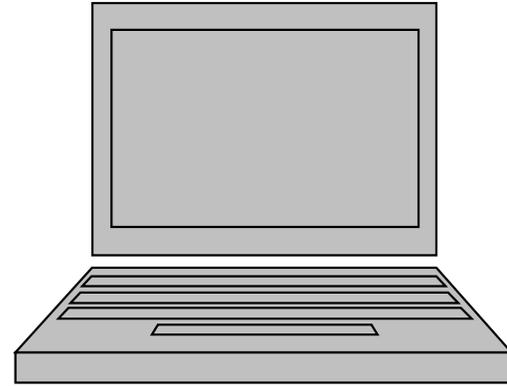
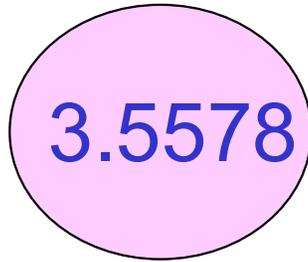
➤ istruzione di **assegnazione**

➤ istruzione di **lettura da dispositivo esterno di input**

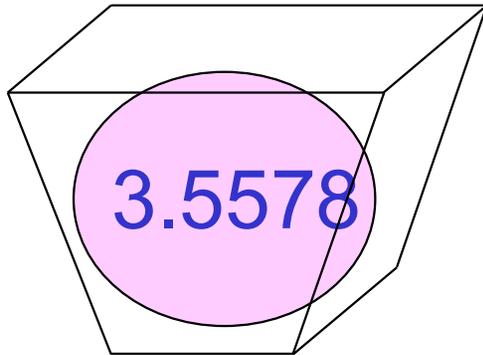
## istruzione di **lettura da dispositivo esterno**

associa a una **variabile** un **valore** fornito da un dispositivo di **input** **durante l'esecuzione** dell'algoritmo

- ✓ una variabile viene **dichiarata**
- ✓ **non** le viene associato alcun valore attraverso **assegnazione**
- ✓ quando l'algoritmo viene **eseguito**, il **valore viene digitato sul dispositivo esterno di input** (la tastiera)
- ✓ il **valore digitato** viene associato alla variabile
- ✓ a questo punto, la variabile è stata **definita**



digitare 3.5578



**lato**

**lato** è una variabile  
di tipo **float**

## istruzione di lettura da dispositivo esterno

l'istruzione deve specificare

- ✓ la **variabile** (destinazione) a cui deve essere assegnato il valore, o un insieme di variabili a ognuna delle quali deve essere assegnato un (diverso) valore

una convenzione può essere:

```
read (<variabili> ) ;
```

**ATTENZIONE:** **read** non è una istruzione C

## istruzione di lettura da dispositivo esterno

associa a una **variabile** un **valore** fornito da un dispositivo di **input** **durante l'esecuzione dell' algoritmo**

Esempio:

```
read (lato) ;
```

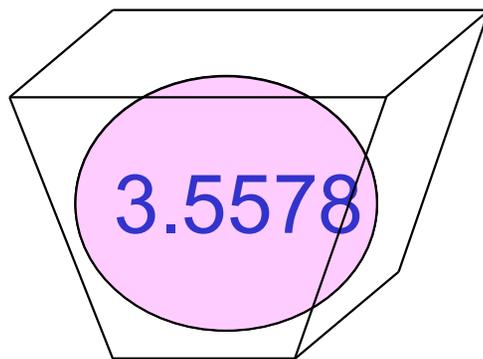
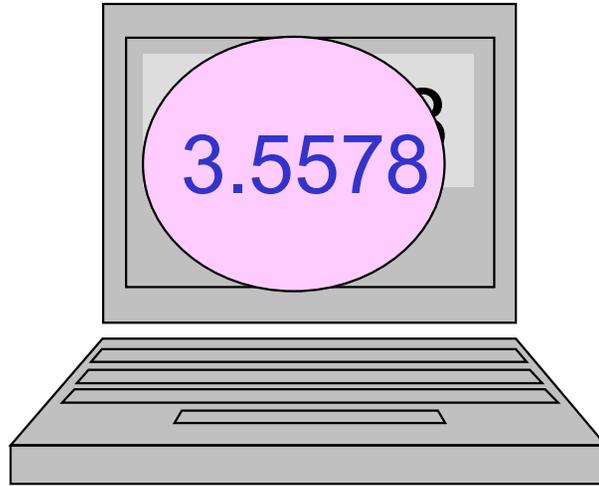
```
read (raggio, circonferenza) ;
```

- ✓ associa a **lato** il dato che viene trasmesso dal dispositivo esterno (digitato sulla tastiera e seguito da **enter**)
- ✓ associa a **raggio** e **circonferenza** i due dati trasmessi dal dispositivo esterno (digitati consecutivamente sulla tastiera e seguiti da **enter**)

## istruzione di **scrittura su dispositivo esterno**

determina l'invio sul **dispositivo standard di uscita**, generalmente lo schermo, del valore associato alle variabili che compaiono nella lista **<variabili>**

- ✓ si **accede** al valore associato a una variabile
- ✓ il valore viene **visualizzato** sul dispositivo esterno (lo schermo)
- ✓ il valore della variabile rimane **inalterato**



**lato**

**lato** è una variabile  
di tipo **float**

## istruzione di **scrittura su dispositivo esterno**

l'istruzione deve specificare

- ✓ la **variabile** il cui valore deve essere visualizzato sul dispositivo esterno

```
printf (<variabili> ) ;
```

## istruzione di scrittura su dispositivo esterno

Esempio:

```
printf (lato) ;
```

```
printf (raggio, circonferenza) ;
```

```
printf (" raggio = ", raggio) ;
```

- ✓ visualizza il valore associato alla variabile **lato**
- ✓ visualizza, in sequenza, il valore associato alla variabile **raggio** e il valore associato alla variabile **circonferenza**
- ✓ visualizza la stringa **raggio =** seguita dal valore associato alla variabile **raggio**

**problema:** sviluppare un algoritmo per calcolare la circonferenza di un cerchio, di cui è noto il raggio

l'algoritmo ha

- ✓ **un dato di input** (il valore del raggio)
- ✓ **un dato di output** (il valore della circonferenza)

l'algoritmo fa uso della famosa formula (chiusa) per il calcolo della circonferenza  $c$  di un cerchio, noto il suo raggio  $r$

$$c = 2 \pi r$$

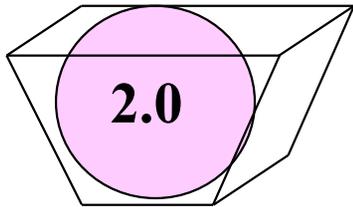
**problema:** sviluppare un algoritmo per calcolare la circonferenza di un cerchio, di cui è noto il raggio

**dato di input:** il valore del raggio (variabile **raggio**)

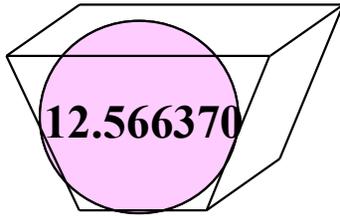
**dato di output:** il valore della circonferenza  
(variabile **circonferenza**)

```
float raggio, circonferenza;  
const float pi_greco = 3.1415926;  
read (raggio) ;  
circonferenza = 2.0*pi_greco*raggio ;  
printf (circonferenza) ;
```

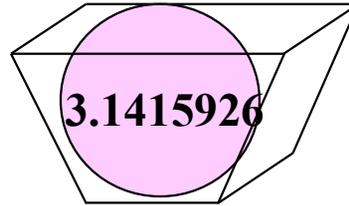
```
float raggio, circonferenza;  
const float pi_greco = 3.1415926;  
read (raggio) ;  
circonferenza = 2.0*pi_greco*raggio ;  
printf (circonferenza) ;
```



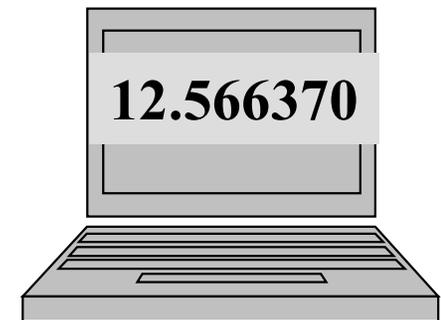
raggio



circonferenza



pi\_greco



si digita 2.0