

Titolo unità didattica: [Approccio incrementale](#)

[06]

Titolo modulo : [Ancora sul calcolo di somme e prodotti](#) [02-T]

Sviluppo di algoritmi per il calcolo di sommatorie e produttorie

Argomenti trattati:

- ✓ algoritmo incrementale per sommatorie generiche
- ✓ algoritmo incrementale per il fattoriale

Prerequisiti richiesti: [P1-06-01-T](#)

richiamo

costruire la soluzione attraverso
incrementi successivi
della difficoltà del problema

approccio incrementale

```
int somma_naturali (int n) {  
    int i, somma;  
    somma = 0 ;  
    for (i=; i <= n; i++) {  
        somma = somma + i ;  
    }  
    return somma ;  
}
```

problema:

calcolo della sommatoria

$$v = \sum_{i=1}^n \frac{1}{i^2} = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \dots + \frac{1}{n^2}$$

dato di input: il numero n (variabile n)

dato di output: il numero v (variabile **somma**)

costrutto ripetitivo: **for**

operazione ripetuta (al generico passo i):

sommare l'addendo i -simo (cioè $1/i^2$)

alla somma degli $(i-1)$ addendi precedenti

$$v = \sum_{i=1}^n \frac{1}{i^2} = 1 + \frac{1}{2^2} + \frac{1}{3^2} + \dots + \frac{1}{n^2}$$

n
addizioni e
2n
prodotti
(molt., div)

```
float somma_quad_rec (int n)
```

```
{  
  int i;  
  float somma;  
  somma = 0 ;  
  for (i=1; i <= n; i++) {  
    somma = somma + 1/((float) i^2);  
  }  
  return somma ;  
}
```

il calcolo del quadrato di un numero si effettua mediante una moltiplicazione

problema:

calcolo della sommatoria generica

$$\sum_{i=1}^n \text{addendo}_i = \text{addendo}_1 + \text{addendo}_2 + \text{addendo}_3 + \dots + \text{addendo}_n$$

dato di input: il numero n (variabile n)

dato di output: il valore della sommatoria (variabile **somma**)

costrutto ripetitivo: **for**

operazione ripetuta (al generico passo i):

sommare l'addendo i -simo (da **calcolare**)

alla somma degli $(i-1)$ addendi precedenti

```
float sommatoria (int n) {  
    int i;  
    float somma ;  
    somma = 0 ;  
    for (i=1; i <= n; i++) {  
        somma = somma + valore di addendo;  
    }  
    return somma ;  
}
```

espressione, function,.... per il calcolo del valore dell' *i*-simo addendo della sommatoria

```
float sommatoria (int n) {  
    int i;  
    float somma, addendo ;  
    somma = 0 ;  
    for (i=1; i <= n; i++) {  
        addendo = valore di addendo; ;  
        somma = somma + addendo ;  
    }  
    return somma ;  
}
```

espressione, function,.... per il calcolo del valore dell' i -simo addendo della sommatoria

problema: calcolo del fattoriale
di un intero

$$n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$$

$$0! = 1, 1! = 1$$

dato di input: il numero n (variabile n)

dato di output: il numero $n!$ (variabile $fatt$)

costrutto ripetitivo: for

operazione ripetuta (al generico passo i):
moltiplicare il numero i (il fattore i -simo
dei prodotti) per il **prodotto degli** ($i-1$)
numeri precedenti



variabile
fatt

$$n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$$

$$0! = 1, 1! = 1$$

```
int fattoriale (int n) {  
    int fatt ;  
    fatt = 1 ;  
    if (n>1) {  
        for (i=2; i <= n; i++)  
            { fatt = fatt * i ; }  
    }  
    return fatt ;  
}
```

n-1
prodotti

sequenza computazionale (**n=5**)

fatt=1; fatt=1*2; fatt=2*3; fatt=6*4;
fatt=24*5

$$n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$$

$$0! = 1, 1! = 1$$

```
int fattoriale (int n) {  
  int fatt ;  
  fatt = 1 ;  
  if (n>1) {  
    for (i=2; i <= n; i++)  
      { fatt = fatt * i ; }  
  }  
  return fatt ;  
}
```

n-1
prodotti

la sequenza computazionale (in questo caso operazioni di prodotto) è costituita da **n-1** prodotti

l'algoritmo ha un **costo** di **n-1** prodotti