

## TIPOLOGIA DI PROBLEMI: SERIE O SUCCESSIONE

“Dati i primi 5 numeri interi strettamente positivi (a partire da 1) mostrare a video la loro somma”

### Prime Osservazioni

1. Il **numero di valori** (in questo caso la **costante 5**) della SERIE o SUCCESSIONE si definisce **DIMENSIONE** o **LUNGHEZZA**

2. La **dimensione minima** da considerare per poter parlare di SERIE o SUCCESSIONE è **1** (un elemento ci deve essere per forza....qui ce ne sono in totale 5)

Quindi la nostra **SERIE o SUCCESSIONE** è costituita dai seguenti valori:

:

**5 valori**  
┌───────────┴───────────┐  
1   2   3   4   5

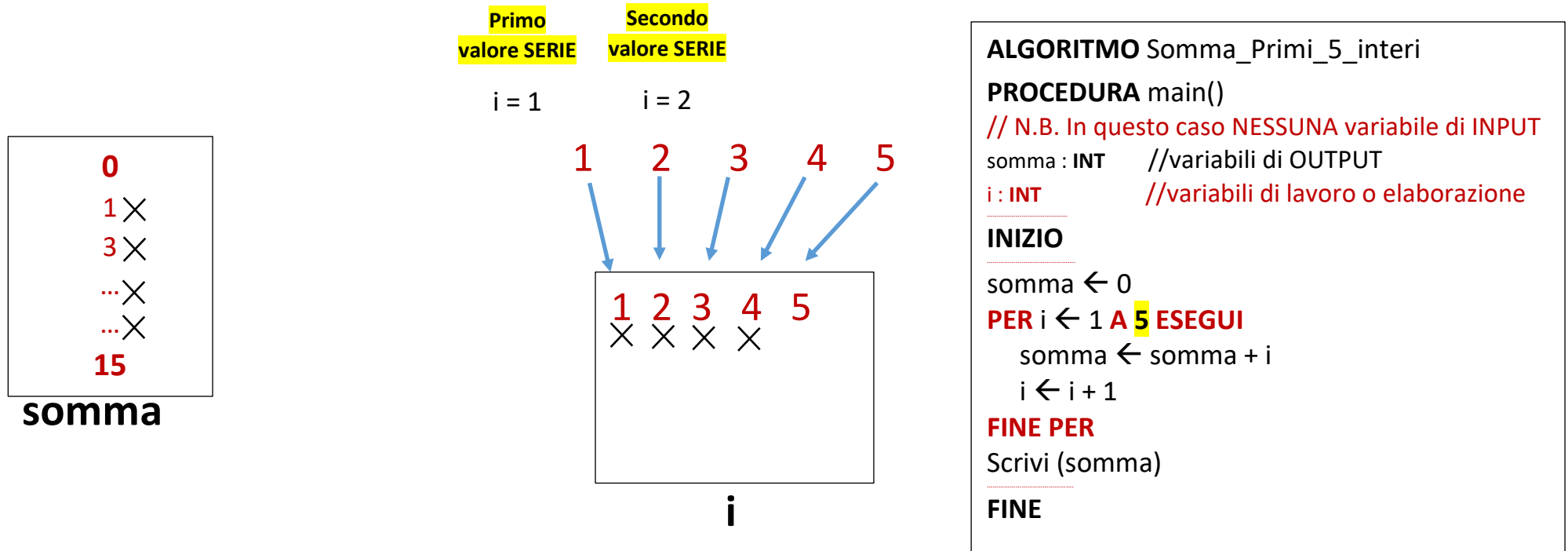
Dobbiamo trovare un algoritmo che risolvendo il problema posto, mostri a video il valore

$$1 + 2 + 3 + 4 + 5$$

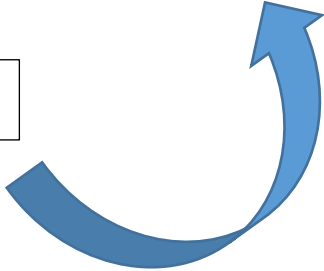
ossia in questo caso, con questi valori deve mostrare a video il valore **15**

Come fare? Ipotesi di modello rappresentativo per la risoluzione del problema...

...questo è un **caso particolare** perché i **valori da sommare** coincidono esattamente con i valori **dell'indice** (o **contatore**) che dovrebbe gestire l'istruzione **ciclica** (facendolo partire proprio da **1**).....quindi:



Trattasi di CICLO ENUMERATIVO



## TABELLE DEI DATI

1

<b>DATI DI INPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

2

<b>DATI DI OUTPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>somma</b>	<b>INT</b>	<b>STATICA</b>	<b>somma = 15</b>	Valore contenente la somma dei valori della SERIE (o SUCESSIONE) dei PRIMI <b>5 numeri interi</b> (a partire da 1) da visualizzare a video n.b QUESTA VOLTA E' LA SOMMA ma potrebbe cambiare a seconda della tipologia di problemi assegnati

3

<b>DATI DI ELABORAZIONE o DI LAVORO DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>i</b>	<b>INT</b>	<b>STATICA</b>	$1 \leq i \leq 5 + 1$ ossia $(i \geq 1) \text{ AND } (i \leq 5 + 1)$	Variabile contenete il contatore che gestisce il ciclo sui valori della SERIE (o SUCESSIONE) n.b. E' SEMPRE UN VALORE INTERO

## PSEUDODODIFICA

**ALGORITMO** Somma\_Prime\_5\_interi

**PROCEDURA** main()

// N.B. In questo caso **NESSUNA** variabile di INPUT

somma : INT //variabili di OUTPUT

i : INT //variabili di lavoro o elaborazione

**INIZIO**

```
.....  
somma ← 0  
PER i ← 1 A 5 ESEGUI  
    somma ← somma + i  
    i ← i + 1  
FINE PER  
Scrivi (somma)
```

```
.....  
somma ← 0  
PER i ← 5 INDIETRO A 1 ESEGUI  
    somma ← somma + i  
    i ← i - 1  
FINE PER  
Scrivi (somma)
```

**CICLO PER**

**FINE**

```
.....  
somma ← 0  
i ← 1  
MENTRE (i ≤ 5) ESEGUI  
    somma ← somma + i  
    i ← i + 1  
FINE PER  
Scrivi (somma)  
.....
```

```
.....  
somma ← 0  
i ← 5  
MENTRE (i ≥ 1) ESEGUI  
    somma ← somma + i  
    i ← i - 1  
FINE PER  
Scrivi (somma)  
.....
```

**CICLO MENTRE**

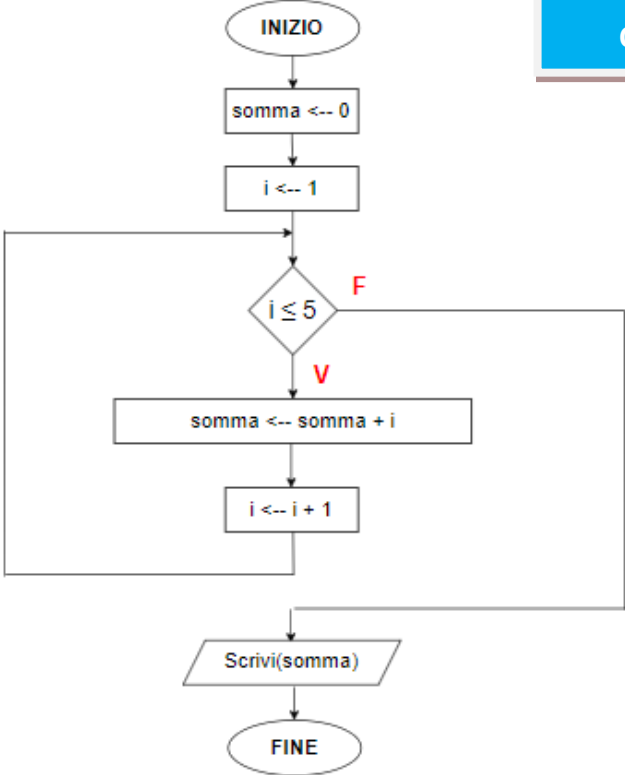
```
.....  
somma ← 0  
i ← 1  
RIPETI  
    somma ← somma + i  
    i ← i + 1  
FINCHE' (i ≥ 5)  
Scrivi (somma)  
.....
```

```
.....  
somma ← 0  
i ← 5  
RIPETI  
    somma ← somma + i  
    i ← i + 1  
FINCHE' (i ≤ 1)  
Scrivi (somma)  
.....
```

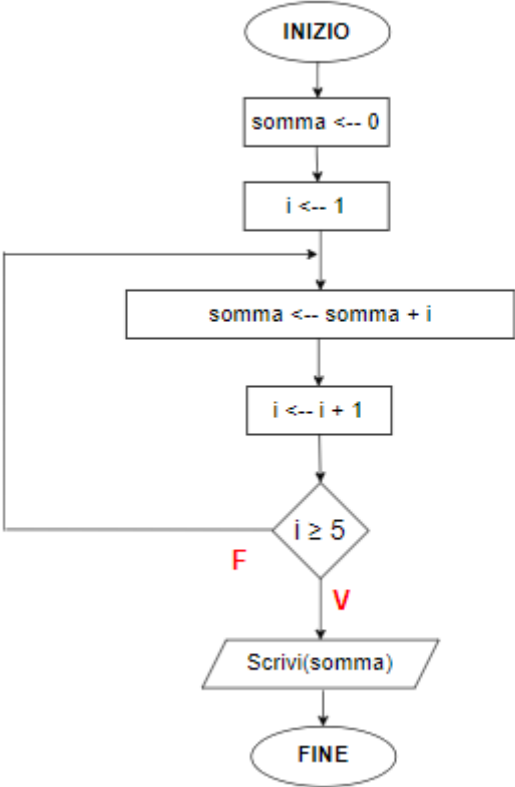
**CICLO RIPETI**

# FLOWCHART

## CICLI PER e MENTRE



## CICLO RIPETI



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

int main(int argc, char*argv[])
{
    /* dati di input */

    /* dati di output */
    int somma;

    /* dati di lavoro o elaborazione */
    int i;

    /* Esamino i valori della SUCCESSIONE uno per uno sommandoli opportunamente*/
    somma = 0;
    for(i = 1; i <= 5; i++)
    {
        /* aggiornno la SOMMA degli elementi della SUCCESSIONE */
        somma = somma + i;
    }

    /* Visualizzo la SOMMA della SUCCESSIONE */
    printf("\nLa SOMMA dei valori della SUCCESSIONE e': ");
    printf("%d", somma);
    printf("\n\n");
    system("PAUSE");
    return 0;
}
```

**GENERALIZZAZIONE DEL PROBLEMA PRECEDENTE:**

Dati i primi **n** numeri interi strettamente positivi (a partire da 1), mostrare a video la loro somma

TABELLE DEI DATI

1

DATI DI INPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>n</b>	<b>INT</b>	<b>STATICA</b>	<b><math>n \geq 1</math></b>	Lunghezza o Dimensione della serie o successione (coincide con il numero complessivo degli elementi)

2

DATI DI OUTPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>somma</b>	<b>INT</b>	<b>STATICA</b>	<b>somma <math>\geq 1</math></b>	Valore contenente la somma dei valori della SERIE (o SUCCESSIONE) dei primi <b>n</b> numeri interi (a partire da 1) da visualizzare a video

3

DATI DI ELABORAZIONE o DI LAVORO DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>i</b>	<b>INT</b>	<b>STATICA</b>	<b><math>1 \leq i \leq n + 1</math></b> ossia <b><math>(i \geq 1) \text{ AND } (i \leq n + 1)</math></b>	Variabile contenete il contatore che gestisce il ciclo sui valori della SERIE (o SUCCESSIONE) <i>n.b. E' SEMPRE UN VALORE INTERO</i>

## PSEUDODODIFICA

**ALGORITMO** Somma\_Prime\_n\_interi

**PROCEDURA** main()

n : INT // variabili di INPUT

somma : INT //variabili di OUTPUT

i : INT //variabili di lavoro o elaborazione

**INIZIO**

**RIPETI**

Leggi (n)

**FINCHE'** (n ≥ 1)

somma ← 0

**PER** i ← 1 **A** n **ESEGUI**

somma ← somma + i

i ← i + 1

**FINE PER**

Scrivi (somma)

CICLO PER

**FINE**

.....

**RIPETI**

Leggi (n)

**FINCHE'** (n ≥ 1)

somma ← 0

i ← 1

**MENTRE** (i ≤ n) **ESEGUI**

somma ← somma + i

i ← i + 1

**FINE PER**

Scrivi (somma)

.....

CICLO MENTRE

.....

**RIPETI**

Leggi (n)

**FINCHE'** (n ≥ 1)

somma ← 0

i ← 1

**RIPETI**

somma ← somma + i

i ← i + 1

**FINCHE'** (i ≥ n)

Scrivi (somma)

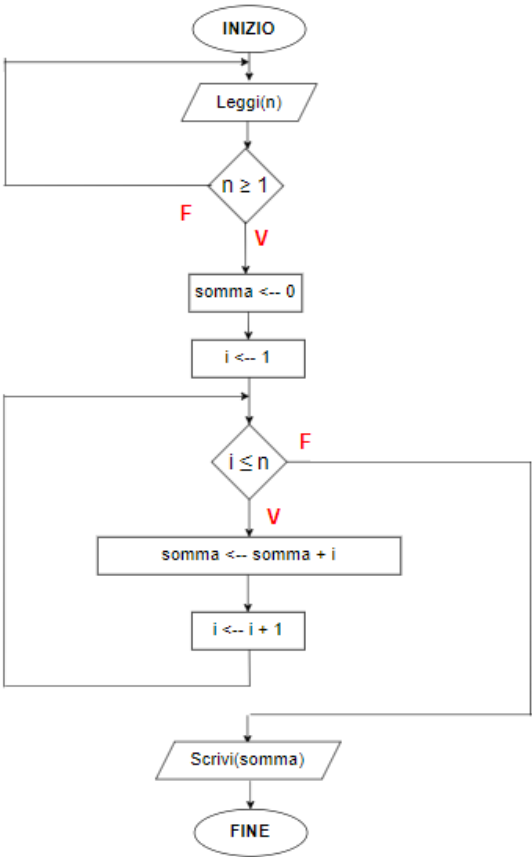
.....

CICLO RIPETI

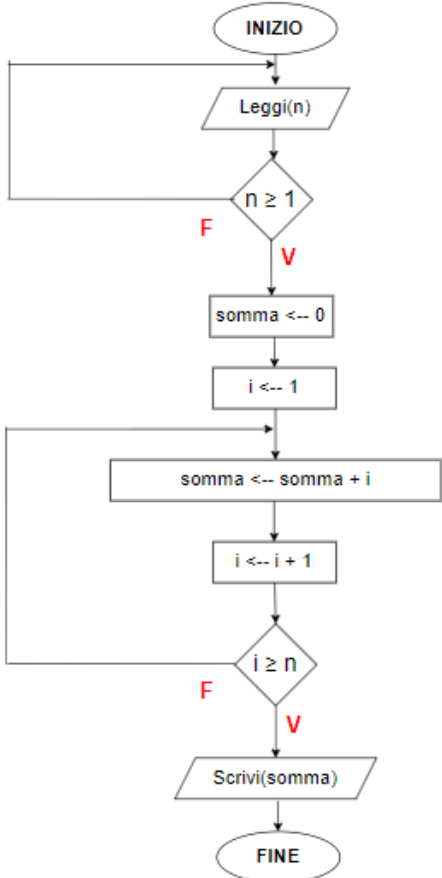


# FLOWCHART

## CICLI PER e MENTRE



## CICLO RIPETI



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

int main(int argc, char*argv[])
{
    /* dati di input */
    int n;
    /* dati di output */
    int somma;
    /* dati di lavoro o elaborazione */
    int i;

    /* Leggo e controllo la dimensione della SERIE o SUCCESSIONE*/
    do
    {
        scanf("%d", &n);
    }
    while (n < 1);
    /* Esamino i valori della SUCCESSIONE uno per uno sommandoli opportunamente*/
    somma = 0;
    for(i = 1; i <= n; i++)
    {
        /* aggiorno la SOMMA degli elementi della SUCCESSIONE */
        somma = somma + i;
    }

    /* Visualizzo la SOMMA della SUCCESSIONE */
    printf("\nLa SOMMA dei valori della SUCCESSIONE e': ");
    printf("%d", somma);
    printf("\n\n");
    system("PAUSE");
    return 0;
}
```

### ULTERIORE GENERALIZZAZIONE DEL PROBLEMA PRECEDENTE:

**Dati  $n$  numeri interi (qualsiasi), mostra a video la loro somma**

## TIPOLOGIA DI PROBLEMI: SERIE O SUCCESSIONE DI VALORI

Del tipo “Dati un certo numero di valori interi (oppure reali oppure caratteri ma anche booleani)..... FARE QUALCOSA

**NOTA BENE:** non molti di voi se ne sono accorti ma anche il problema P7 (max tra tre numeri interi) altro non era che

**UN CASO PARTICOLARE** (sono 3 valori) di un problema più generale (max tra un certo numero di valori interi)

**Esempio di Problema che tratta SERIE o SUCCESSIONE di valori:**

**“Dati  $n$  numeri interi mostrare a video la loro somma”**

### Prime Osservazioni

1. Il **numero di valori** (indicato genericamente con la variabile  $n$ ) della SERIE o SUCCESSIONE si definisce **DIMENSIONE** o **LUNGHEZZA**
2. La **dimensione minima** da considerare per poter parlare di SERIE o SUCCESSIONE è 1 (quindi  $n \geq 1$ )
3. Non conoscendo a PRIORI il numero di valori ossia la dimensione di una SERIE o SUCCESSIONE (ricordiamo che  $n$  è una variabile) NON POSSO USARE UNA VARIABILE PER CIASCUN VALORE DELLA SERIE.

Devo cercare di generalizzare il problema....ma come fare?

4. Si utilizza un'unica variabile (sempre la stessa) per ospitare di volta in volta il valore corrente della SERIE o SUCCESSIONE

# “Dati **n** numeri interi (QUALSIASI) mostrare a video la loro somma”

Esempio: Supponiamo che la nostra SERIE o SUCCESSIONE sia formata dalla sequenza di **7** valori INTERI:

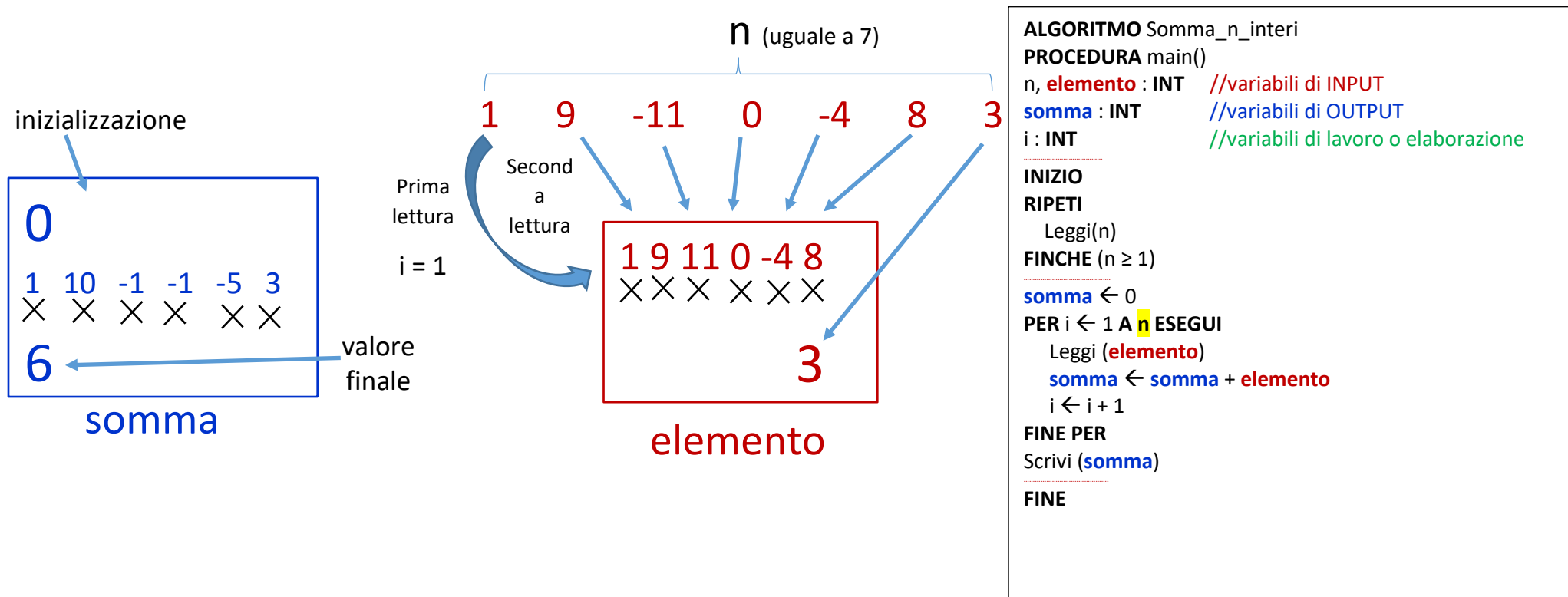
1 9 -11 0 -4 8 3

Dobbiamo trovare un algoritmo che risolvendo il problema posto, mostri a video il valore

$$1 + 9 + (-11) + 0 + (-4) + 8 + 3$$

ossia in questo caso, con questi valori deve mostrare a video il valore **6**

Come fare? Ipotesi di modello rappresentativo per la risoluzione del problema...



## TABELLE DEI DATI

1

<b>DATI DI INPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>n</b>	<b>INT</b>	<b>STATICA</b>	<b><math>n \geq 1</math></b>	Lunghezza o dimensione (ossia il numero di valori che ne fanno parte) delle SERIE (o SUCCESSIONE) immesso da tastiera <i>n.b. E' SEMPRE UN VALORE INTERO</i>
<b>elemento</b>	<b>INT</b>	<b>STATICA</b>	<b>TUTTI</b>	Valore corrente della SERIE (o SUCCESSIONE) immesso da tastiera <i>n.b QUESTA VOLTA E' INTERO ma potrebbe cambiare a seconda della tipologia di problemi assegnati</i>

2

<b>DATI DI OUTPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>somma</b>	<b>INT</b>	<b>STATICA</b>	<b>TUTTI</b>	Valore contenente la somma dei valori della SERIE (o SUCCESSIONE) da visualizzare a video <i>n.b QUESTA VOLTA E' LA SOMMA ma potrebbe cambiare a seconda della tipologia di problemi assegnati</i>

3

<b>DATI DI ELABORAZIONE o DI LAVORO DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>i</b>	<b>INT</b>	<b>STATICA</b>	<b><math>1 \leq i \leq n + 1</math> ossia <math>(i \geq 1) \text{ AND } (i \leq n + 1)</math></b>	Variabile contenete il contatore che gestisce il ciclo sui valori della SERIE (o SUCCESSIONE) <i>n.b. E' SEMPRE UN VALORE INTERO</i>

**ALGORITMO** Somma\_n\_Interi\_Qualsiasi

**PROCEDURA** main()

/\* dati di input \*/

elemento, n: **INT**

/\* dati di output \*/

somma: **INT**

/\* dati di lavoro o elaborazione \*/

i: **INT**

**INIZIO**

/\* Leggo e controllo la dimensione della SUCCESIONE \*/

**RIPETI**

  Scrivi("Inserire la dimensione della SUCCESIONE: ")

  Leggi(n)

**FINCHE'** (n ≥ 1)

/\* Esamino i valori della SUCCESIONE uno per uno sommandoli opportunamente\*/

somma ← 0

**PER** i ← 1 **A** n **ESEGUI**

  /\* leggo il generico elemento della SUCCESIONE \*/

  Scrivi("inserire il valore della SUCCESIONE: ")

  Leggi(elemento)

  /\* aggiorno la SOMMA degli elementi della SUCCESIONE \*/

  somma ← somma + elemento

  i ← i + 1

**FINE PER**

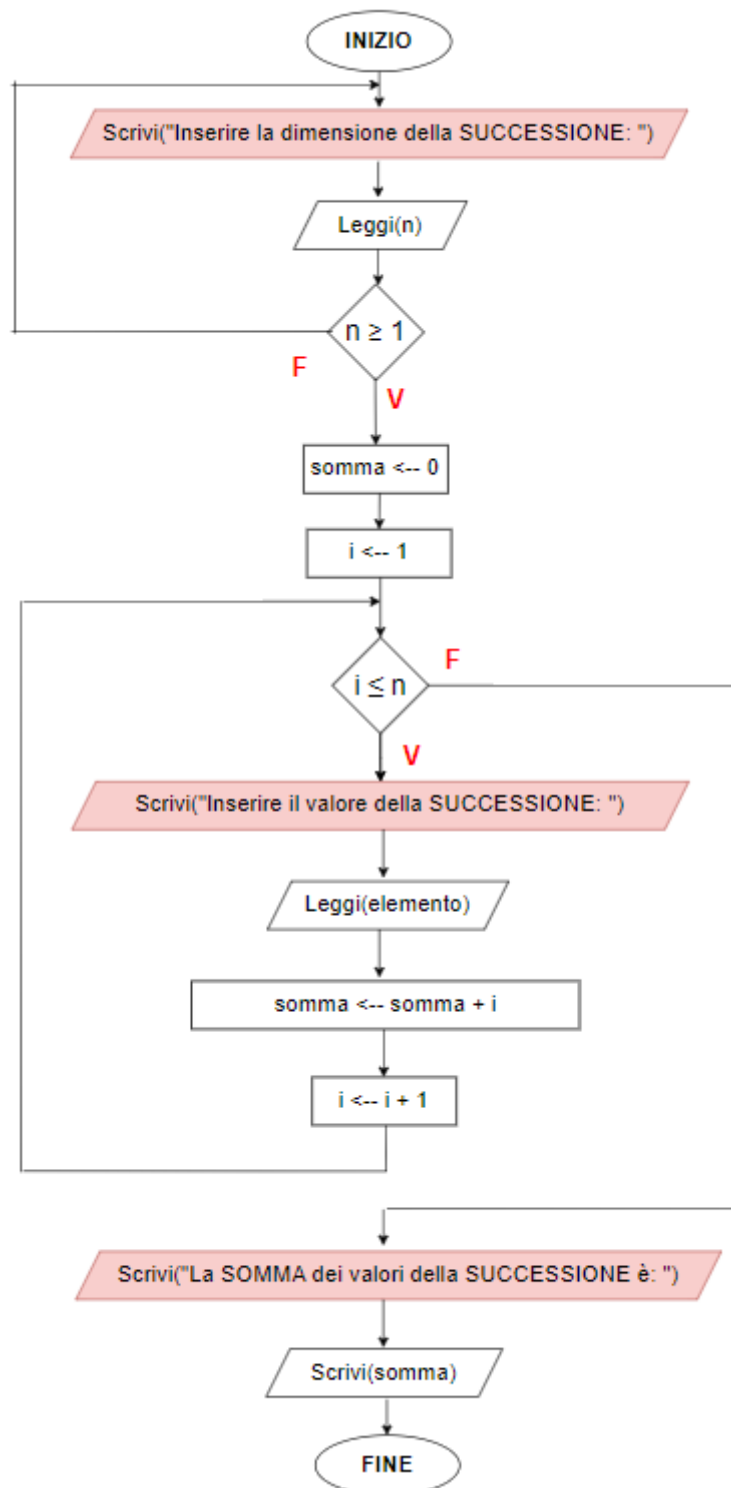
/\* Visualizzo la SOMMA della SUCCESIONE \*/

Scrivi("La SOMMA dei valori della SUCCESIONE è: ")

Scrivi(somma)

**FINE**

# FLOWCHART



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

int main(int argc, char*argv[])
{
    /* dati di input */
    int elemento, n;

    /* dati di output */
    int somma;

    /* dati di lavoro o elaborazione */
    int i;

    /* Leggo e controllo la dimensione della SUCCESSIONE */
    do
    {
        printf("Inserire la dimensione della SUCCESSIONE: ");
        scanf("%d", &n);
    }
    while (n < 1);
    /* Esamino i valori della SUCCESSIONE uno per uno sommandoli opportunamente*/
    somma = 0;
    for(i = 1; i <= n; i++)
    {
        /* leggo il generico elemento della SUCCESSIONE */
        printf("Inserire un valore della SUCCESSIONE: ");
        scanf("%d", &elemento);
        /* aggiorno la SOMMA degli elementi della SUCCESSIONE */
        somma = somma + elemento;
    }

    /* Visualizzo la SOMMA della SUCCESSIONE */
    printf("\nLa SOMMA dei valori della SUCCESSIONE e': ");
    printf("%d", somma);
    printf("\n\n");
    system("PAUSE");
    return 0;
}
```



## TIPOLOGIA DI PROBLEMI: SERIE O SUCCESSIONE DI VALORI CAMBIA QUALCOSA.....HANNO QUACHE COSA IN OIYUO.

Esempio di Problema che tratta SERIE o SUCCESSIONE di valori CONDIZIONATA:

**“Dati  $n$  numeri interi qualsiasi mostrare a video la loro somma arrestandosi immediatamente al primo valore nullo”**

Esempio:

Se Ipotizziamo che la serie abbia dimensione  $n = 4$  e che sia formata dai numeri

**-1 , 2, -3 , 4**

- allora la somma degli elementi sarà pari a  $(-1 + 2 - 3 + 4) = 2$

Se Ipotizziamo che la serie sia

**-1 , 2, 0**

allora essa avrà dimensione  $n = 2$  e sarà formata dai soli primi due numeri (poi c'è lo zero).

In questo caso allora la somma degli elementi da mostrare sarà pari  $(-1 + 2) = 1$  poiché è presente nella serie il **valore 0** (valore scelto per l'arresto).

**(N.B. PER OVVIE RAGIONI l'eventuale valore 0 non deve essere contato per esprimere la lunghezza della serie)**

### Prime Osservazioni

**Oltre a tutte le osservazioni già fatte in precedenza per le SERIE bisogna aggiungerne un'altra FONDAMENTALE:**

5. In questo tipo di algoritmo NON SONO SEMPRE SICURO di dover esaminare tutti i valori della SERIE ossia il CRITERIO per ARRESTARE IL PROCESSO RISOLUTIVO non è UNICO .....

MI DEVO FERMARE SE.....

Sono arrivato a fine SERIE perché non c'è alcun valore 0 (quindi ho esaminato tutti i valori della serie)

Ho interrotto il mio processo risolutivo perché ho incontrato uno 0 (quindi mi sono devo arrestare nell'esaminare i valori della serie)

## TABELLE DEI DATI

1

<b>DATI DI INPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>n</b>	<b>INT</b>	<b>STATICA</b>	<b><math>n \geq 1</math></b>	Lunghezza o dimensione (ossia il numero di valori che ne fanno parte) delle SERIE (o SUCCESSIONE) immesso da tastiera <i>n.b. E' SEMPRE UN VALORE INTERO</i>
<b>elemento</b>	<b>INT</b>	<b>STATICA</b>	<b>TUTTI</b>	Valore corrente della SERIE (o SUCCESSIONE) immesso da tastiera <i>n.b QUESTA VOLTA E' INTERO ma potrebbe cambiare a seconda della tipologia di problemi assegnati</i>

2

<b>DATI DI OUTPUT DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>somma</b>	<b>INT</b>	<b>STATICA</b>	<b>TUTTI</b>	Valore contenente la somma dei valori della SERIE (o SUCCESSIONE) da visualizzare a video <i>n.b QUESTA VOLTA E' LA SOMMA ma potrebbe cambiare a seconda della tipologia di problemi assegnati</i>
<b>pos</b>	<b>INT</b>	<b>STATICA</b>	<b><math>1 \leq pos \leq n</math> ossia <b><math>(pos \geq 1) \text{ AND } (pos \leq n)</math></b></b>	Variabile contenente la posizione del primo elemento NULLO all'interno della SUCCESSIONE(o SERIE) <i>n.b. E' SEMPRE UN VALORE INTERO</i>

3

<b>DATI DI ELABORAZIONE o DI LAVORO DEL PROBLEMA PRINCIPALE (PROCEDURA MAIN)</b>				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
<b>i</b>	<b>INT</b>	<b>STATICA</b>	<b><math>1 \leq i \leq n + 1</math> ossia <b><math>(i \geq 1) \text{ AND } (i \leq n + 1)</math></b></b>	Variabile contenete il contatore che gestisce il ciclo sui valori della SERIE (o SUCCESSIONE) <i>n.b. E' SEMPRE UN VALORE INTERO</i>
<b>trovato</b>	<b>BOOL</b>	<b>STATICA</b>	<b>VERO o FALSO</b>	Variabile che indica se presente il <b>valore 0</b> all'interno della SERIE (o SUCCESSIONE) <i>n.b. E' SEMPRE UN VALORE INTERO</i>

**ALGORITMO** Somma\_Condizionata\_n\_Interi

**PROCEDURA** main()

/\* dati di input \*/

elemento, n: **INT**

/\* dati di output \*/

somma, pos: **INT**

/\* dati di lavoro o elaborazione \*/

i: **INT**

trovato : **BOOL**

**INIZIO**

/\* Leggo e controllo la dimensione della SUCESSIONE \*/

**RIPETI**

  Scrivi("Inserire la dimensione della SUCESSIONE: ")

  Leggi(n)

**FINCHE'** (n ≥ 1)

/\* Esamino i valori della SUCESSIONE uno per uno sommandoli opportunamente \*/

/\* fermandoli nel caso appaia un valore pari a 0 \*/

somma ← 0

trovato ← **FALSO**

pos ← 0

i ← 1

**MENTRE** (i ≤ n) AND (trovato = **FALSO**) **ESEGUI**

  /\* leggo il generico elemento della SUCESSIONE \*/

  Scrivi("inserire un valore SUCESSIONE è: ")

  Leggi(elemento)

**SE** (elemento ≠ 0)

**ALLORA**

      /\* aggiorno la SOMMA degli elementi della SUCESSIONE solo se i valori sono diversi da zero\*/

      somma ← somma + elemento

      i ← i + 1

**ALTRIMENTI**

      /\* se nella SUCESSIONE è presente 0 mio devo fermare \*/

      trovato ← **VERO**

      pos = i;

**FINE SE**

**FINE MENTRE**

/\* valuto l'eventuale presenza dello 0 all'interno della SUCESSIONE \*/

**SE** (trovato = **VERO**)

**ALLORA**

    Scrivi("Il valore 0 E' presente nella SUCESSIONE in posizione")

    Scrivi(pos)

**SE** (pos >= 2)

**ALLORA**

        /\* Visualizzo la SOMMA della SUCESSIONE \*/

        Scrivi("La SOMMA parziale dei valori della SUCESSIONE fino allo zero e': ")

        Scrivi(somma)

**ALTRIMENTI**

        Scrivi("questo caso il processo risolutivo si arresta e non ha senso parlare di somma!!");

**FINE SE**

**ALTRIMENTI**

    Scrivi("Il valore 0 NON E' presente nella SUCESSIONE!")

  /\* Visualizzo la SOMMA della SUCESSIONE \*/

  Scrivi("La SOMMA dei valori della SUCESSIONE è: ")

  Scrivi(somma)

**FINE SE**

**FINE**

```
#include <stdio.h>
#include <stdlib.h>
#define FALSO      0
#define VERO       1
int main(int argc, char*argv[])
{
    /* dati di input */
    int elemento, n;
    /* dati di output */
    int somma, pos;
    /* dati di lavoro o elaborazione */
    int i, trovato;

    /* Leggo e controllo la dimensione della SUCCESSIONE */
    do
    {
        printf("Inserire la dimensione della SUCCESSIONE: ");
        scanf("%d", &n);
    }
    while (n < 1);
    /* Esamino i valori della SUCCESSIONE uno per uno sommandoli opportunamente*/
    somma = 0;
    trovato = FALSO;
    i = 1;
    while ((i <= n) && (trovato == FALSO))
    {
        /* leggo il generico elemento della SUCCESSIONE */
        printf("Inserire un valore della SUCCESSIONE: ");
        scanf("%d", &elemento);
        if (elemento != 0)
        {
            /* aggiorno la SOMMA degli elementi della SUCCESSIONE solo se i valori sono diversi da zero*/
            somma = somma + elemento;
            i = i + 1;
        }
        else
        {
            /* se nella SUCCESSIONE è presente 0 mio devo fermare */
            trovato = VERO;
            pos ← i
        }
    }
    /* valuto l'eventuale presenza dello 0 all'interno delle SUCCESSIONE */
    if (trovato == VERO)
    {
        printf("Il valore 0 e' presente nella SUCCESSIONE in posizione: ");
        printf("%d",i);
        if ( i >= 2)
        {
            /* Visualizzo la SOMMA della SUCCESSIONE */
            printf("\nLa SOMMA parziale dei valori della SUCCESSIONE fino allo zero e': ");
            printf("%d", somma);
        }
        else
        {
            printf("\nIn questo caso il processo risolutivo si arresta e non ha senso parlare
di somma!!");
        }
    }
}
```

```
else
{
printf("Il valore 0 NON E' presente nella SUCCESSIONE!");
/* Visualizzo la SOMMA della SUCCESSIONE */
printf("La SOMMA dei valori della SUCCESSIONE è: ");
printf("%d",somma);
}
printf("\n\n");
system("PAUSE");
return 0;
}
```