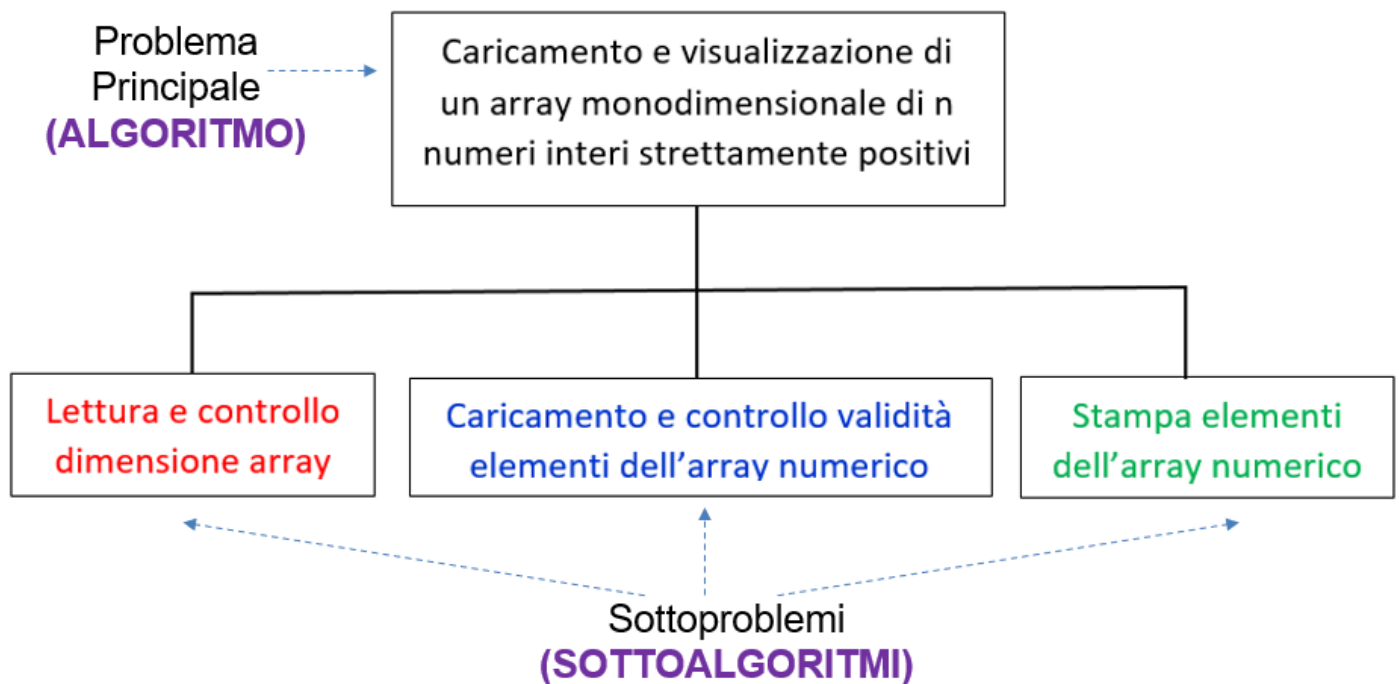


PROBLEMA: caricamento e visualizzazione array di n numeri interi strettamente positivi (SOTTOPROGRAMMI)

N.B. Utilizzando la la METODOLOGIA di PROGETTAZIONE TOP-DOWN il problema assegnato potrebbe essere scomposto nei seguenti sottoproblemi:



Grazie alla metodologia di progettazione top-down, abbiamo dunque individuato i seguenti **SOTTOPROGRAMMI** (procedura o funzione) da implementare, che si devono occupare delle seguenti azioni:

- 1) la **lettura** ed il **controllo** della **dimensione dell'array**;
- 2) il **caricamento** (con eventuale controllo del valore) dei suoi elementi;
- 3) la **stampa** dei suoi elementi.

1 **FUNZIONE** Leggi_Dim () : INT
Prototipo o segnatura se **FUNZIONE**

2 **PROCEDURA** Carica_V (VAL n : INT, REF v : ARRAY [MAXDIM] DI INT)
Prototipo o segnatura se **PROCEDURA**

3 **PROCEDURA** Visualizza_V (VAL n : INT, VAL v : ARRAY [MAXDIM] DI INT)
Prototipo o segnatura se **PROCEDURA**

TABELLE DEI DATI

DATI DI INPUT DEL PROBLEMA PRINCIPALE PROCEDURA main()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

DATI DI OUTPUT DEL PROBLEMA PRINCIPALE PROCEDURA main()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

DATI DI ELABORAZIONE o DI LAVORO DEL PROBLEMA PRINCIPALE PROCEDURA main()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
MAXDIM	INT	STATICA	10	Massimo numero di elementi dell'array monodimensionale v
v	ARRAY[MAXDIM] DI INT	STATICA	$v[i] > 0$	Vettore monodimensionale di interi positivi da acquisire da tastiera
n	REAL	STATICA	$1 \leq n \leq \text{MAXDIM}$ ossia $(n \geq 1) \text{ AND } (n \leq \text{MAXDIM})$	Dimensione del vettore di interi positivi

ALGORITMO Array_SottoProgrammi

MAXDIM 10

PROCEDURA main()

/* dati di elaborazione o lavoro */

v : **ARRAY**[MAXDIM] **DI INT**

n : **INT**

INIZIO

/* CALL alla **FUNZIONE** Leggi_Dim() */

n ← Leggi_Dim();

/* CALL alla **PROCEDURA** Carica_V() */

Carica_V (n, v)

/* CALL alla **PROCEDURA** Visualizza_V() */

Visualizza_V (n, v)

RITORNA

FINE

DATI DI INPUT DEL SOTTOPROBLEMA: FUNZIONE Leggi_Dim()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
n	INT	STATICA	$1 \leq n \leq \text{MAXDIM}$ ossia $(n \geq 1) \text{ AND } (n \leq \text{MAXDIM})$	Dimensione del vettore di interi positivi da restituire nel nome della funzione stessa

DATI DI OUTPUT DEL SOTTOPROBLEMA: FUNZIONE Leggi_Dim()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
MAXDIM	INT	STATICA	10	Massimo numero di elementi dell'array monodimensionale v

DATI DI ELABORAZIONE o DI LAVORO DEL SOTTOPROBLEMA: FUNZIONE Leggi_Dim()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

FUNZIONE Leggi_Dim () : INT

n : INT

INIZIO

/ Leggo e controllo la dimensione dell'array */*

RIPETI

Scrivi("Inserisci la dimensione dell'array: ")

 Leggi(n)

SE (n < 1) **OR** (n > MAXDIM)

ALLORA

Scrivi("ERRORE: la dimensione deve essere compresa tra 1 e")

Scrivi(MAXDIM)

FINE SE

FINCHE' (n ≥ 1) **AND** (n ≤ MAXDIM)

RITORNA (n)

FINE

DATI DI INPUT DEL SOTTOPROBLEMA: PROCEDURA Carica_V()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
n	INT	STATICA	$1 \leq n \leq \text{MAXDIM}$ ossia $(n \geq 1) \text{ AND } (n \leq \text{MAXDIM})$	Primo parametro FORMALE, PASSATO PER VALORE, che conterrà la dimensione del vettore

DATI DI OUTPUT DEL SOTTOPROBLEMA: PROCEDURA Carica_V()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
v	ARRAY[MAXDIM] DI INT	STATICA	$v[i] > 0$	Secondo parametro FORMALE, PASSATO PER RIFERIMENTO che conterrà il vettore che si intende caricare

DATI DI ELABORAZIONE o DI LAVORO DEL SOTTOPROBLEMA: PROCEDURA Carica_V()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
i	INT	STATICA	$1 \leq i \leq n + 1$ ossia $(i \geq 1) \text{ AND } (i \leq n + 1)$	Variabile che conterrà l'indice necessario a navigare gli elementi del vettore v

PROCEDURA Carica_V (**VAL** n : INT, **REF** v : ARRAY[**MAXDIM**] DI INT)

INIZIO

i : INT

/ Leggo e controllo i valori dell'array */*

PER i ← 1 **A** n **ESEGUI**

RIPETI

Scrivi("Inserisci l'elemento dell'array: ")

Leggi(v[i])

SE (v[i] ≤ 0)

ALLORA

Scrivi("ERRORE: l'elemento dell'array deve essere strettamente positivo!")

FINE SE

FINCHE' (v[i] > 0)

 i ← i + 1

FINE PER

RITORNA

FINE

DATI DI INPUT DEL SOTTOPROBLEMA: PROCEDURA Visualizza_V ()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
n	INT	STATICA	$1 \leq n \leq \text{MAXDIM}$ ossia $(n \geq 1) \text{ AND } (n \leq \text{MAXDIM})$	Primo parametro FORMALE, PASSATO PER VALORE, che conterrà la dimensione del vettore
v	ARRAY[MAXDIM] DI INT	STATICA	$v[i] > 0$	Secondo parametro FORMALE, PASSATO PER VALORE che conterrà il vettore che si intende stampare

DATI DI OUTPUT DEL SOTTOPROBLEMA: PROCEDURA Visualizza_V ()				
Nome variabile	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione

DATI DI ELABORAZIONE o DI LAVORO DEL SOTTOPROBLEMA: PROCEDURA Visualizza_V ()				
Nome variabile oppure nome costante	Tipo dati	Tipo Allocazione	Valori ammessi	Descrizione
i	INT	STATICA	$1 \leq i \leq n + 1$ ossia $(i \geq 1) \text{ AND } (i \leq n + 1)$	Variabile che conterrà l'indice necessario a navigare gli elementi del vettore v

PROCEDURA Visualizza_V (**VAL** n : INT, **VAL** v : ARRAY[**MAXDIM**] DI INT)

INIZIO

i : INT

/ Stampa i valori dell'array */*

Scrivi("Gli elementi dell'array sono: ")

PER i ← 1 **A** n **ESEGUI**

 Scrivi(v[i])

 Scrivi(" ")

 i ← i + 1

FINE PER

RITORNA

FINE

ALGORITMO Array_SottoPogrammi

MAXDIM 10

```

PROCEDURA main()
/* dati di elaborazione o lavoro */
v : ARRAY[MAXDIM] DI INT
n : INT
INIZIO
/* CALL alla FUNZIONE Leggi_Dim () */
n ← Leggi_Dim();
/* CALL alla PROCEDURA Carica_V () */
Carica_V (n, v)
/* CALL alla PROCEDURA Visualizza_V () */
Visualizza_V (n, v)
RITORNA
FINE

```

```

FUNZIONE Leggi_Dim () : INT
n : INT
INIZIO
/* Leggo e controllo la dimensione dell'array */
RIPETI
  Scrivi("Inserisci la dimensione dell'array: ")
  Leggi(n)
  SE (n < 1) OR (n > MAXDIM)
    ALLORA
      Scrivi("ERRORE: la dimensione deve essere compresa tra 1 e")
      Scrivi(MAXDIM)
    FINE SE
  FINCHE' (n ≥ 1) AND (n ≤ MAXDIM)
RITORNA (n)
FINE

```

```

PROCEDURA Carica_V (VAL n : INT, REF v : ARRAY[MAXDIM] DI INT)
INIZIO
i : INT
/* Leggo e controllo i valori dell'array */
PER i ← 1 A n ESEGUI
  RIPETI
    Scrivi("Inserisci l'elemento dell'array: ")
    Leggi(v[i])
    SE (v[i] ≤ 0)
      ALLORA
        Scrivi("ERRORE: l'elemento dell'array deve essere strettamente positivo!")
      FINE SE
    FINCHE' (v[i] > 0)
    i ← i + 1
FINE PER
RITORNA
FINE

```

PROCEDURA Visualizza_V (**VAL** n : INT, **VAL** v : ARRAY[MAXDIM] DI INT)

INIZIO

i : INT

/ Stampo i valori dell'array */*

Scrivi("Gli elementi dell'array sono: ")

PER i ← 1 **A** n **ESEGUI**

Scrivi(v[i])

Scrivi(" ")

i ← i + 1

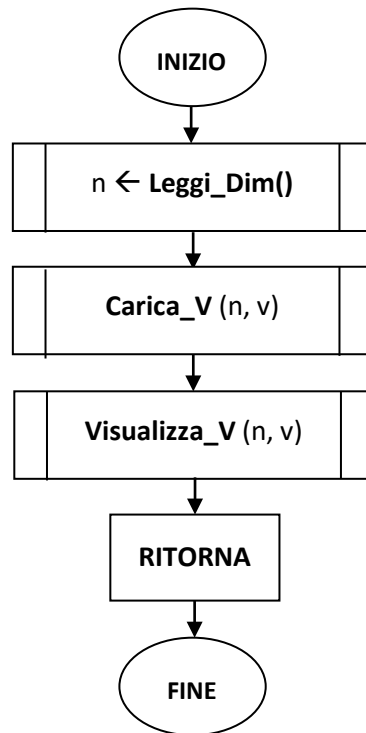
FINE PER

RITORNA

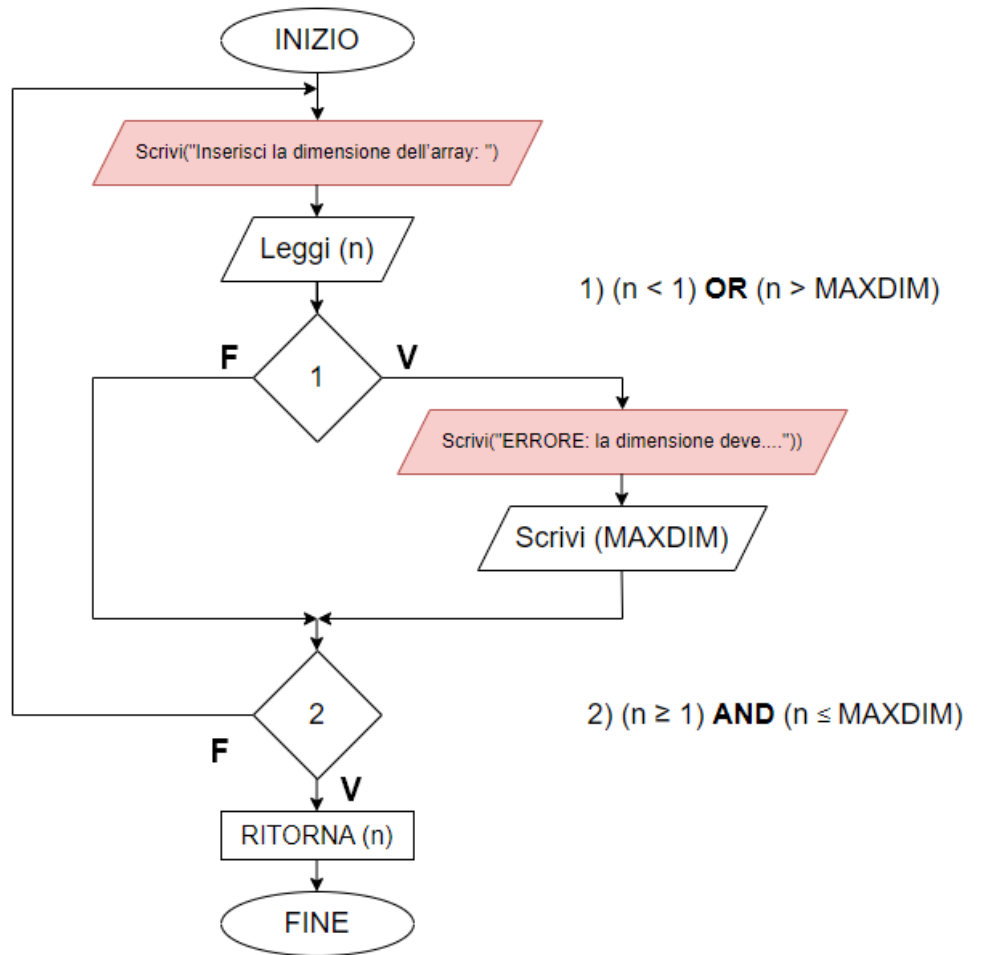
FINE

FLOWCHART

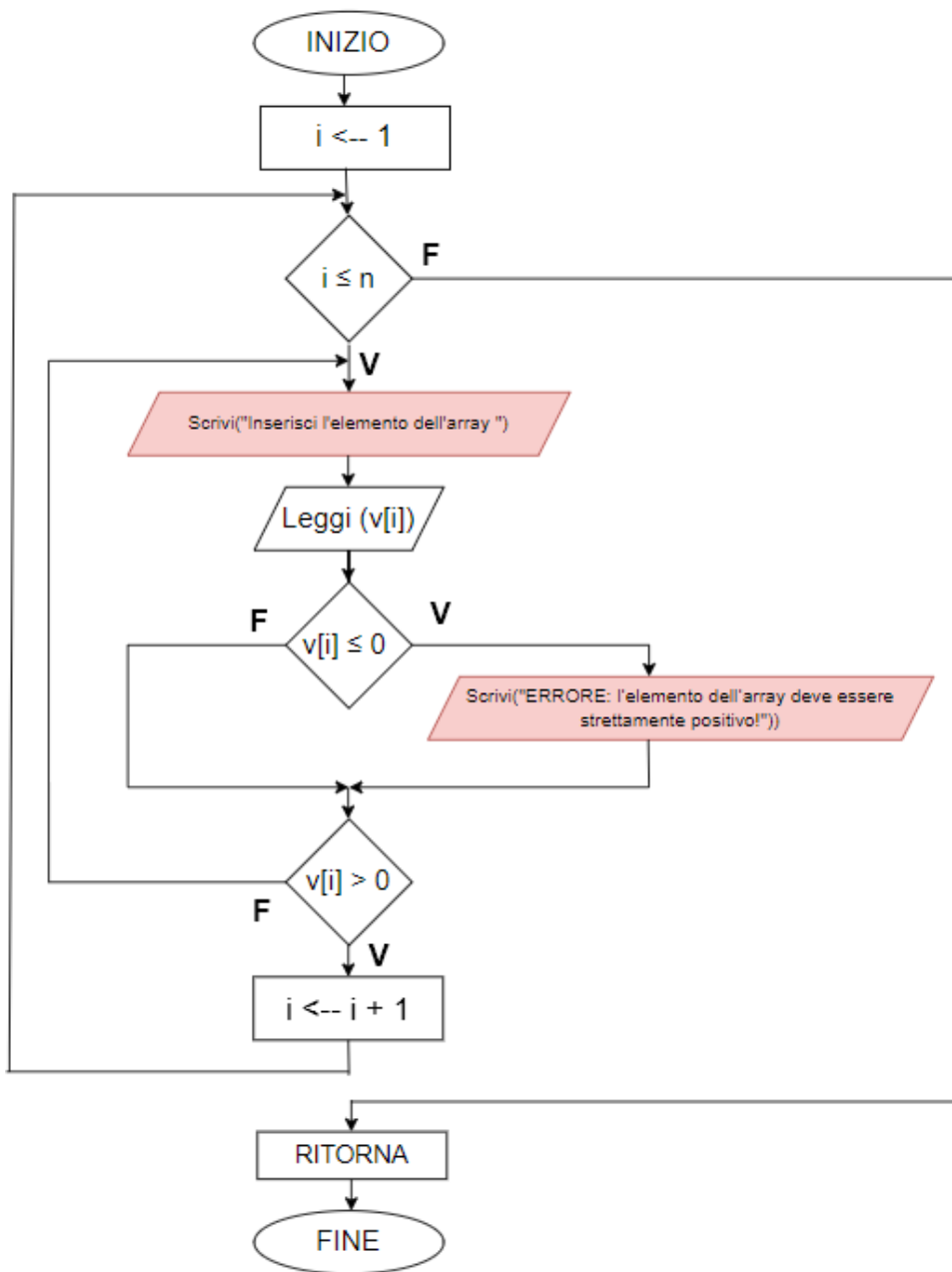
PROCEDURA **main** ()



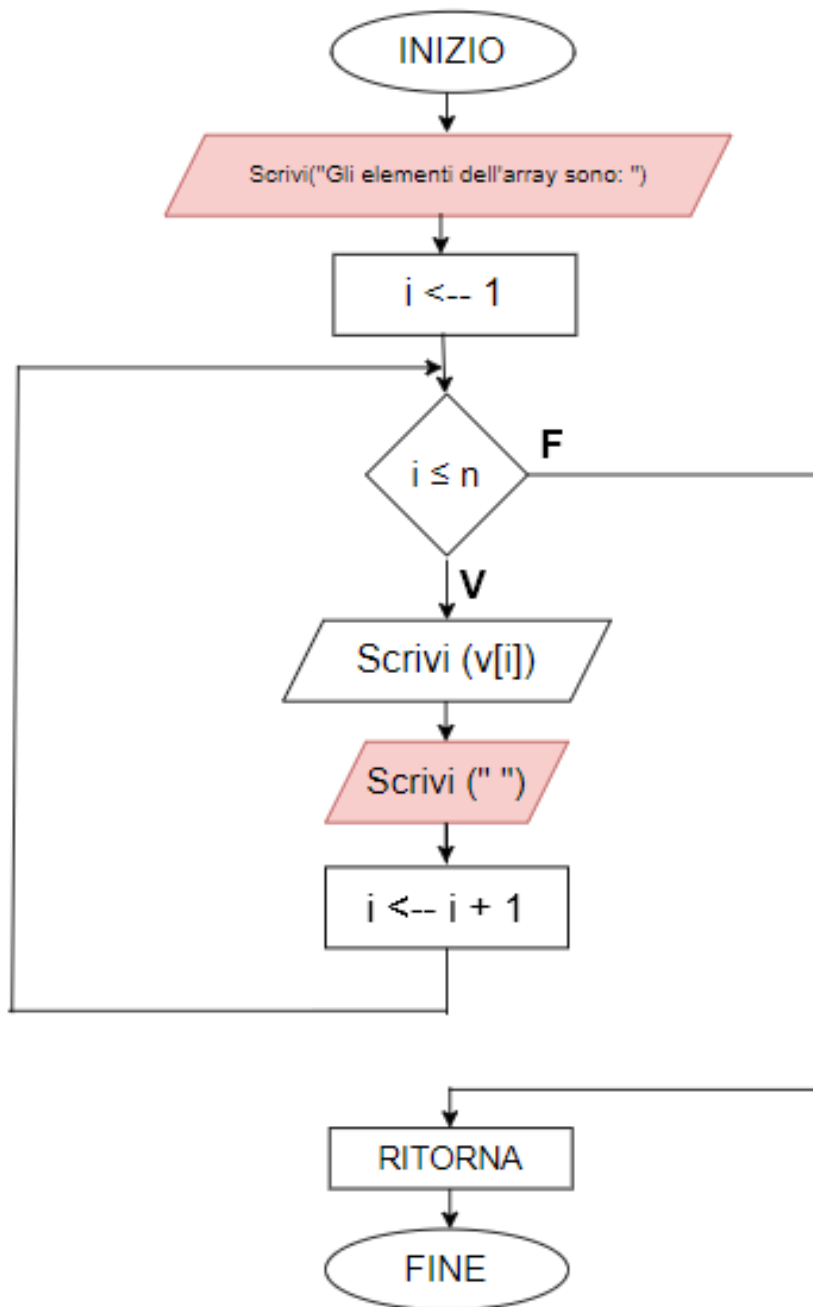
FUNZIONE **Leggi_Dim** ()



PROCEDURA **Carica_V** ()



PROCEDURA **Visualizza_V** ()



```

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

#define MAXDIM 10

// N.B. ESCLUSIVAMENTE nei prototipi può essere omissa il NOME DEI PARAMETRI FORMALI, NON IL TIPO
// definizione del prototipo della funzione Leggi_Dim (teoricamente è una FUNZIONE)
int Leggi_Dim ();
// definizione del prototipo delle funzione Carica_V (teoricamente è una PROCEDURA)
void Carica_V (int n , int v[MAXDIM]);
// definizione dei prototipo delle funzione Visualizza_V (teoricamente è una PROCEDURA)
void Visualizza_V (int n , int v[MAXDIM]);

////////////////////////////////////
// FUNZIONE main() //
////////////////////////////////////
int main(int argc, char*argv[])
{
/* dati di input/output */
int v[MAXDIM];
int n;

/* CALL alla FUNZIONE Leggi_Dim() */
n = Leggi_Dim();
/* CALL alla PROCEDURA Carica_V () */
Carica_V (n, v);
/* CALL alla PROCEDURA Visualizza_V () */
Visualizza_V (n, v);

return 0;
}

////////////////////////////////////
// FUNZIONE Leggi_Dim () // //(teoricamente è una FUNZIONE)
////////////////////////////////////
int Leggi_Dim (void)
{
//dati di input
int n;

/* Leggo e controllo la dimensione dell'array */
do
{
printf ("Inserisci la dimensione dell'array: ");
scanf ("%d", &n);
if ((n < 1) || (n > MAXDIM))
{
printf("ERRORE: la dimensione deve essere compresa tra 1 e ");
printf("%d\n", MAXDIM);
}
}
while ((n < 1) || (n > MAXDIM));

return (n);
}

```

```

////////////////////////////////////
//  FUNZIONE Carica_V ()  // //(teoricamente è una PROCEDURA)
////////////////////////////////////
void Carica_V (int n, int v[MAXDIM])
{
//dati di lavoro
int i;

/* Leggo e controllo i valori dell'array */
for(i = 0; i < n; i++)
{
do
{
printf ("Inserisci l'elemento del vettore: ");
scanf ("%d", &v[i]);
if (v[i] <= 0)
{
printf ("ERRORE: l'elemento dell'array deve essere strettamente positivo!\n");
}
}
while (v[i] <= 0);
}

return;
}

```

```

////////////////////////////////////
//  FUNZIONE Visualizza_V ()  // //(teoricamente è una PROCEDURA)
////////////////////////////////////
void Visualizza_V (int n, int v[MAXDIM])
{
//dati di lavoro
int i;

/* Leggo e controllo i valori dell'array */
for(i = 0; i < n; i++)
{
printf ("%d", v[i]);
printf (" ");
}

return;
}

```