

Introduzione alla programmazione

Esercizi risolti

1 Esercizio

Si determini se il diagramma di flusso rappresentato in Figura 1 è strutturato.

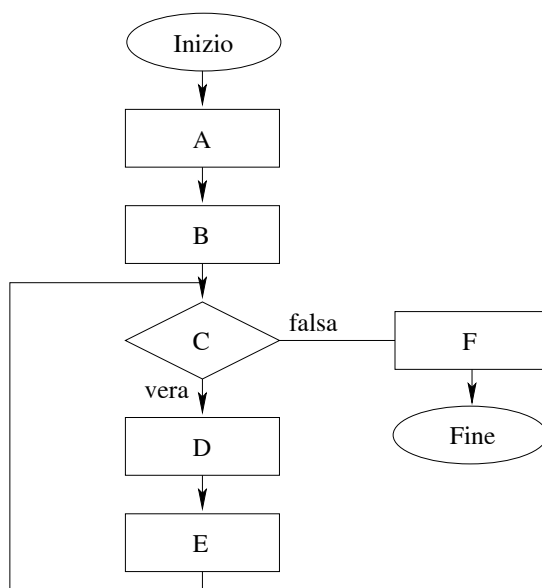


Figura 1: Diagramma di flusso strutturato?

Soluzione

Per determinare se il diagramma di flusso è strutturato è necessario ricondurlo a essere rappresentato mediante una sequenza di blocchi elementari:

- operazioni
- costrutti condizionali **if-then-else** e **if-then**
- costrutti iterativi **while** (o **for**) e **do-while**.

Durante il riconoscimento di tali blocchi è possibile procedere sostituendo a ogni blocco un blocco elementare semplice. In Figura 2 (a) si riconosce la sequenza di blocchi (A+B e D+E). In Figura 2 (b) si riconosce il costrutto iterativo **while** (costituito dai blocchi C e (D+E)). In Figura 2 (c) si riconosce la sequenza di blocchi (A+B), (C+D+E) e F. Il diagramma di flusso finale, essendo costituito da un unico blocco, è sicuramente strutturato.

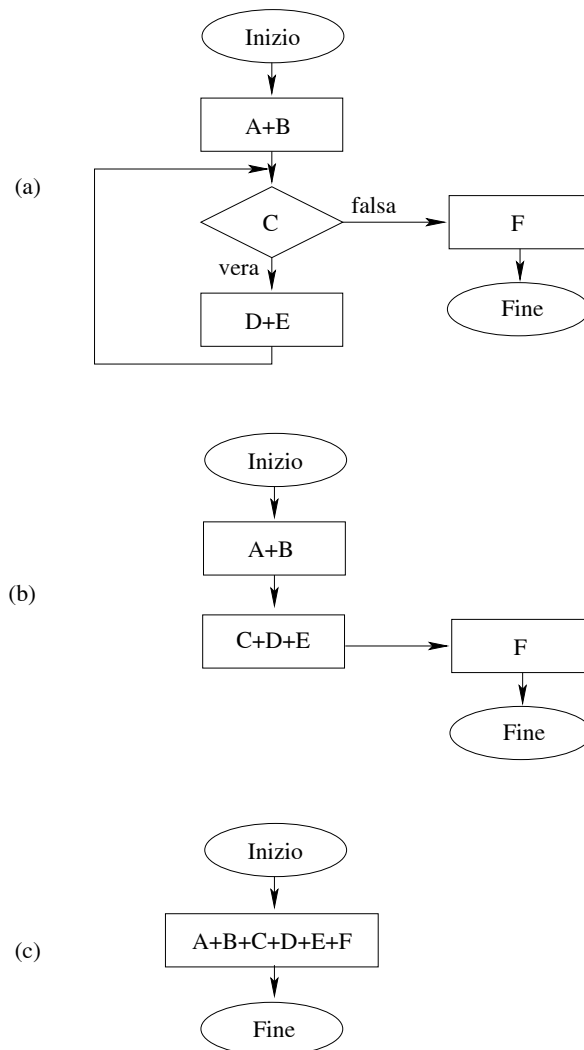


Figura 2: Diagramma di flusso strutturato: soluzione.

2 Esercizio

Si rappresenti mediante diagramma di flusso un algoritmo che:

- legga un numero intero n
- calcoli e visualizzi il valore della ridotta n -esima della serie logaritmica:

$$H(n) = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{n}$$

Esempio Si supponga di introdurre il valore 3. L'algoritmo deve calcolare la somma

$$H(3) = 1 + \frac{1}{2} + \frac{1}{3} = 1.833$$

e visualizzare il valore risultante 1.833.

Soluzione

La soluzione di questo esercizio richiede la stesura di un ciclo per valutare il valore della somma del reciproco dei primi n numeri naturali. La soluzione rappresentata in Figura 3 utilizza un ciclo **while** definito con conteggio in “salita” (ciclo $1 \rightarrow n$). La variabile di controllo del ciclo è i . H è il valore calcolato. Ad ogni iterazione il nuovo valore di H è uguale al vecchio valore di H incrementato del nuovo termine della successione.

3 Esercizio

Si rappresenti mediante diagramma di flusso l'algoritmo in grado di:

- leggere una sequenza di numeri interi sino a introdurre un valore nullo
- sommare tra di loro tutti i valori strettamente positivi e tutti i valori strettamente negativi della sequenza (trascurando quindi i valori nulli)
- visualizzare le due somme calcolate.

Esempio Si supponga di introdurre i seguenti valori:

5 4 -3 1 -2 0

Occorre visualizzare i valori 10 (ottenuto dalla somma $5 + 4 + 1$) e -5 (ottenuto dalla somma $(-3) + (-2)$). Si osservi che il numero di valori letti risulta indefinito a priori, in quanto dipende dall'introduzione di un valore nullo, informazione nota solo in fase di esecuzione.

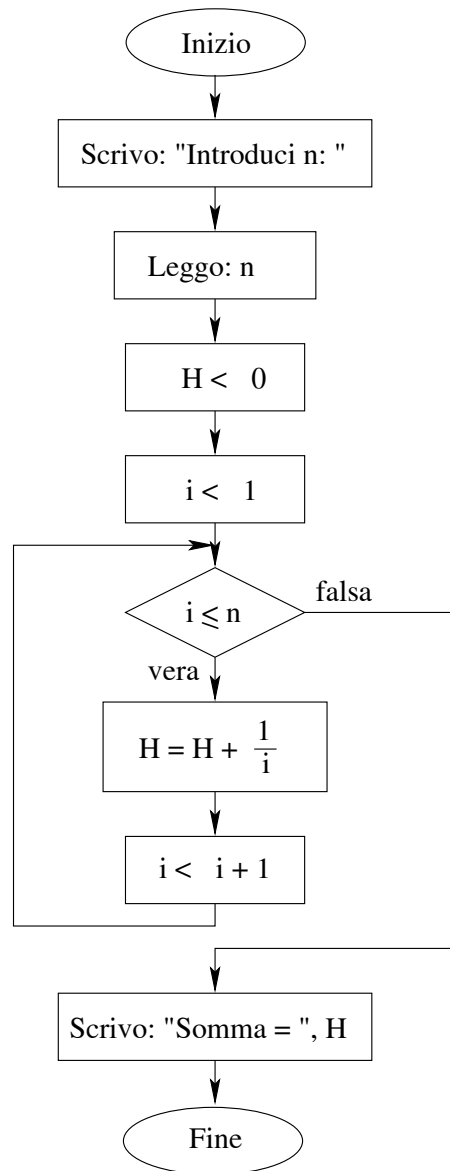


Figura 3: Serie armonica.

Soluzione

La soluzione presentata in Figura 4 è caratterizzata dall'utilizzare un ciclo `do-while`. Questo evita la lettura del primo valore della sequenza di ingresso prima del ciclo. Le variabili `sommaP` e `sommaN` permettono di memorizzare le due somme richieste, ovvero rispettivamente la somma dei numeri positivi e negativi. Si noti come i numeri

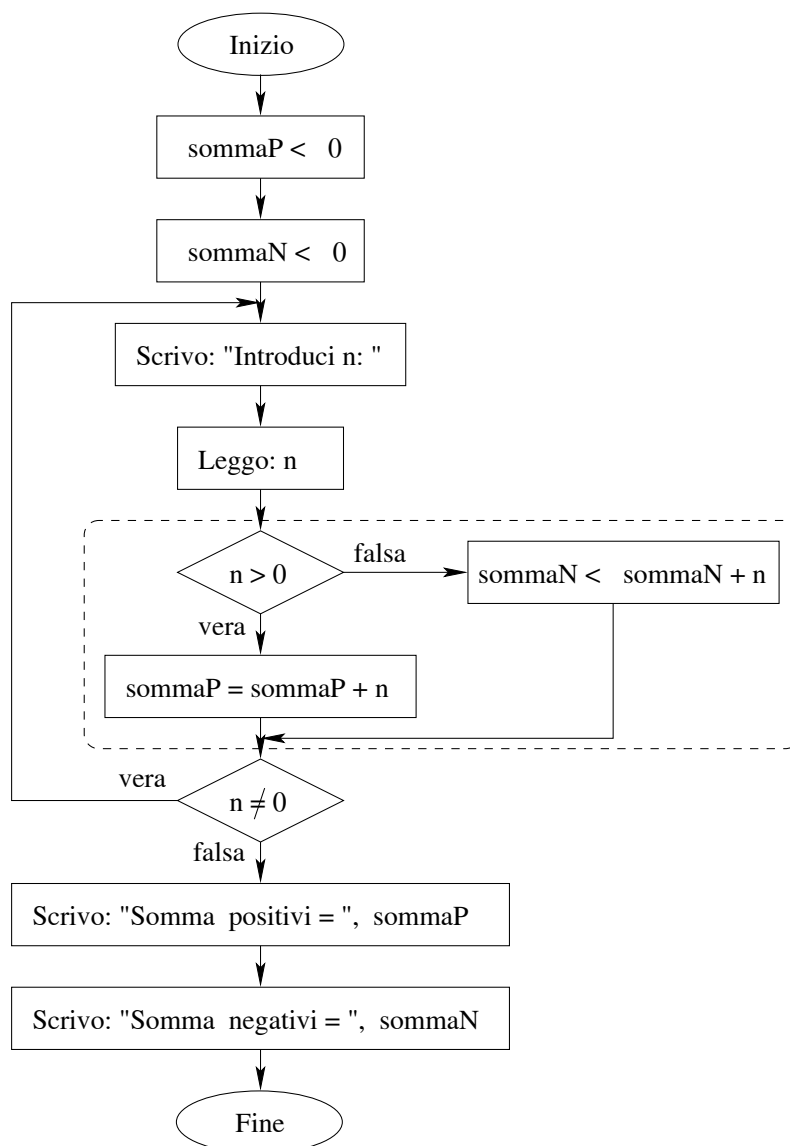


Figura 4: Somma valori positivi e negativi.

nulli vengano, di fatto, sommati insieme ai numeri positivi. Questa operazione non produce però alcun effetto, visto che il loro contributo alla somma è nullo. Volendo comunque evitare questo effetto si può procedere sostituendo la sezione incorniciata

di Figura 4 come indicato in Figura 5 (e lasciando il resto del diagramma inalterato).

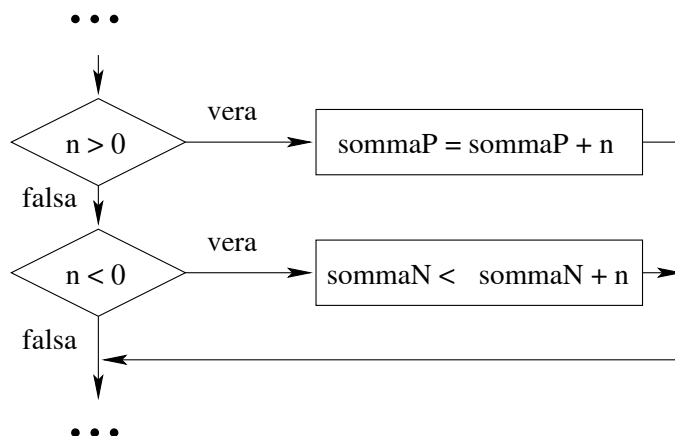


Figura 5: Somma valori positivi e negativi: Controllo più accurato sul valore dei numeri.

4 Esercizio

Realizzare un diagramma di flusso in grado di:

- leggere, da tastiera, una sequenza di 10 numeri interi
- visualizzare la massima differenza (in valore assoluto) tra 2 numeri consecutivi della sequenza. L'algoritmo visualizzi, inoltre, i valori tra cui questa differenza è ottenuta.

Esempio Si supponga di introdurre la seguente sequenza di 10 valori:

8 15 22 123 9 23 45 23 45 2

La massima differenza è ottenuta tra i numeri 123 e 9 ed è pari a 114. Occorre visualizzare i valori 123, 9 e 114.

Soluzione

La soluzione è rappresentata in Figura 6. n_1 e n_2 indicano i due valori letti da tastiera più recentemente e $diff$ il valore assoluto della differenza di tali valori, $diffMax$ rappresenta la differenza massima, ovvero il massimo valore di $diff$ sino a un determinato istante, e $val1$ e $val2$ i valori per cui tale differenza massima si è verificata. Durante la prima iterazione la condizione $i > 1$ è falsa e si procede quindi a copiare n_2 in n_1 e a incrementare il contatore i . In tutte le altre iterazioni

tale condizione è vera e quindi si effettua il calcolo di `diff` e si valuta se tale valore è maggiore o uguale a `diffMax`. In caso affermativo vengono memorizzati i “nuovi” valori di `n1` e `n2` che determinano tale differenza e il nuovo valore di `diffMax`. Si noti che viene controllato se `diff` \geq `diffMax` e non `diff` $>$ `diffMax` in quanto si vuole garantire che `val1` e `val2` assumano valore corretto anche nel caso in cui tutti i valori introdotti da tastiera siano uguali e `diff` sia sempre uguale a zero.

5 Esercizio

Modificare il diagramma di flusso rappresentato in Figura 7 per disegnare le figure di seguito rappresentate (di dimensione variabile):

1. triangoli di dimensione n:

```
*
**
***
****
*****
```

2. triangoli “rovesciati” di dimensione n:

```
*****
****
***
**
*
```

3. quadrati vuoti di dimensione n:

```
*****
*      *
*      *
*      *
*      *
*****
```

4. quadrati composti dalle cifre [0,9] disposte in sequenza:

```
012345
678901
234567
890123
456789
012345
```

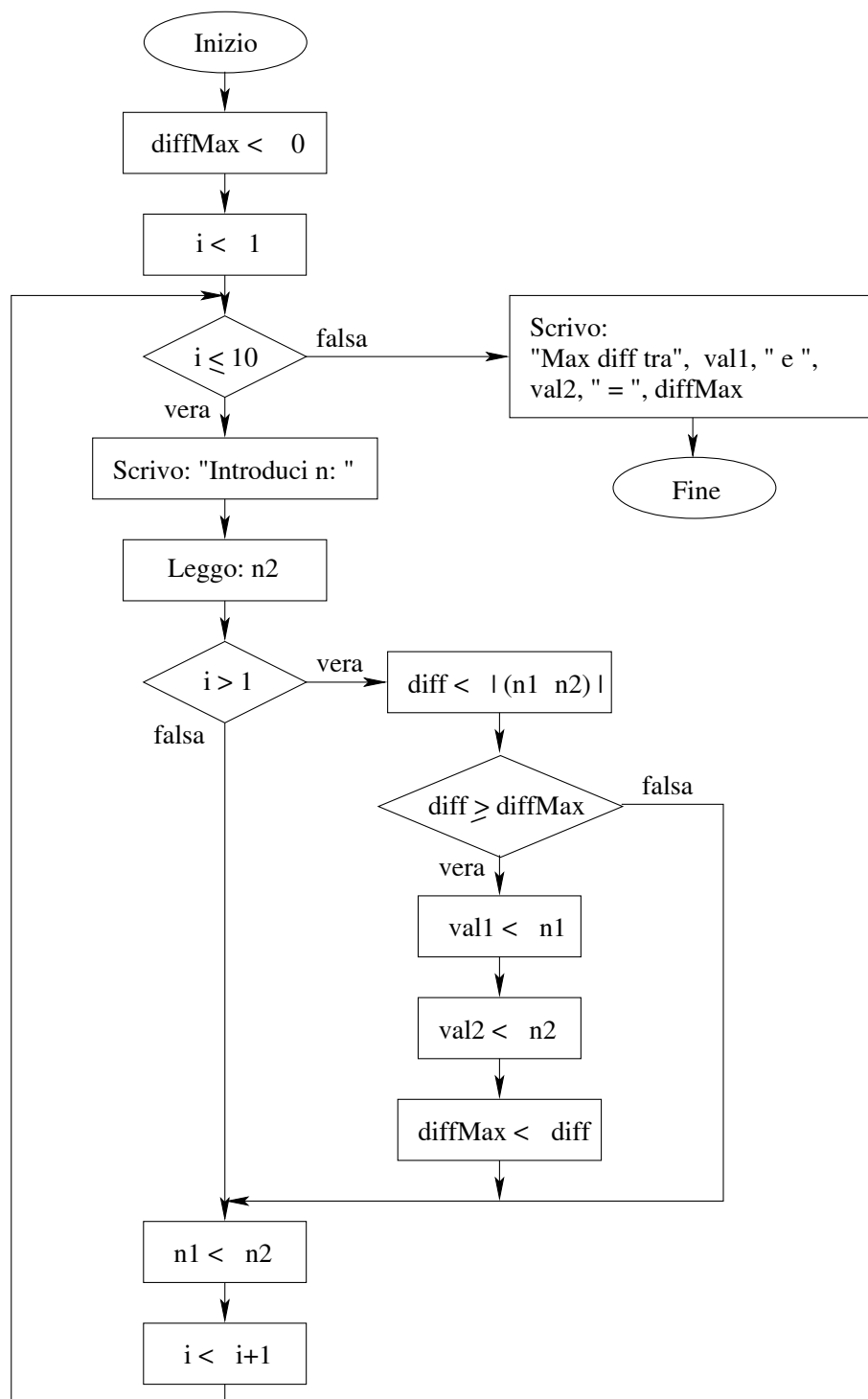


Figura 6: Massima differenza.

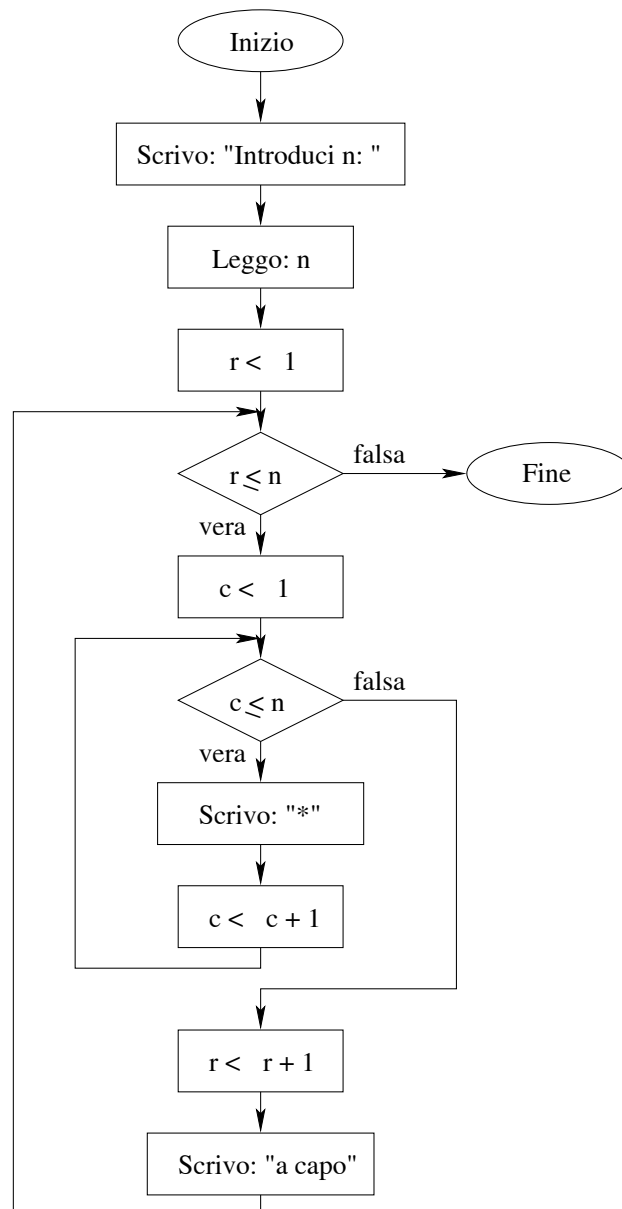


Figura 7: Visualizzazione di un quadrato.

Soluzione 1

Una possibile soluzione è rappresentata in Figura 8. Rispetto al diagramma di flusso rappresentato in Figura 7 le modifiche si limitano alla condizione di terminazione del ciclo interno. Su ogni riga il numero di asterischi è uguale al valore di `r`, ovvero dell'indice che controlla il movimento lungo le righe.

Soluzione 2

Una possibile soluzione è rappresentata in Figura 9. Rispetto al diagramma di flusso rappresentato in Figura 7 le modifiche sono nuovamente minime. Effettuando un ciclo decrescente sulla variabile `r` è possibile procedere come per l'algoritmo Figura 8.

Soluzione 3

La soluzione è rappresentata in Figura 10. Rispetto al diagramma di flusso rappresentato in Figura 7 le modifiche sono più consistenti. Invece di disegnare comunque un carattere '*' si rappresenta tale carattere solo sulla "cornice" del quadrato. Si visualizza invece uno spazio all'interno del quadrato. Per determinare quale simbolo visualizzare si controllano i valori di `r` e `c` per determinare se si è sulla prima riga, sulla prima colonna, sull'ultima colonna o sull'ultima riga.

Soluzione 4

La soluzione è rappresentata in Figura 11. Rispetto al diagramma di flusso rappresentato in Figura 7 occorre visualizzare un valore numerico crescente da 0 a 9 invece del carattere '*'. Si utilizza quindi la variabile `val`, inizializzata a 0 e incrementata dopo ogni visualizzazione sino al valore massimo. Una volta raggiunto il valore massimo a `val` viene riassegnato il valore zero.

6 Esercizio

Si rappresenti mediante diagramma di flusso l'algoritmo in grado di:

- leggere un valore intero n
- disegnare un quadrato di lato n utilizzando:
 - caratteri '*' lungo la diagonale
 - caratteri '-' al di sotto della diagonale
 - caratteri '+' al di sopra della diagonale.

Esempio Sia $n = 5$. Occorre visualizzare la seguente figura:

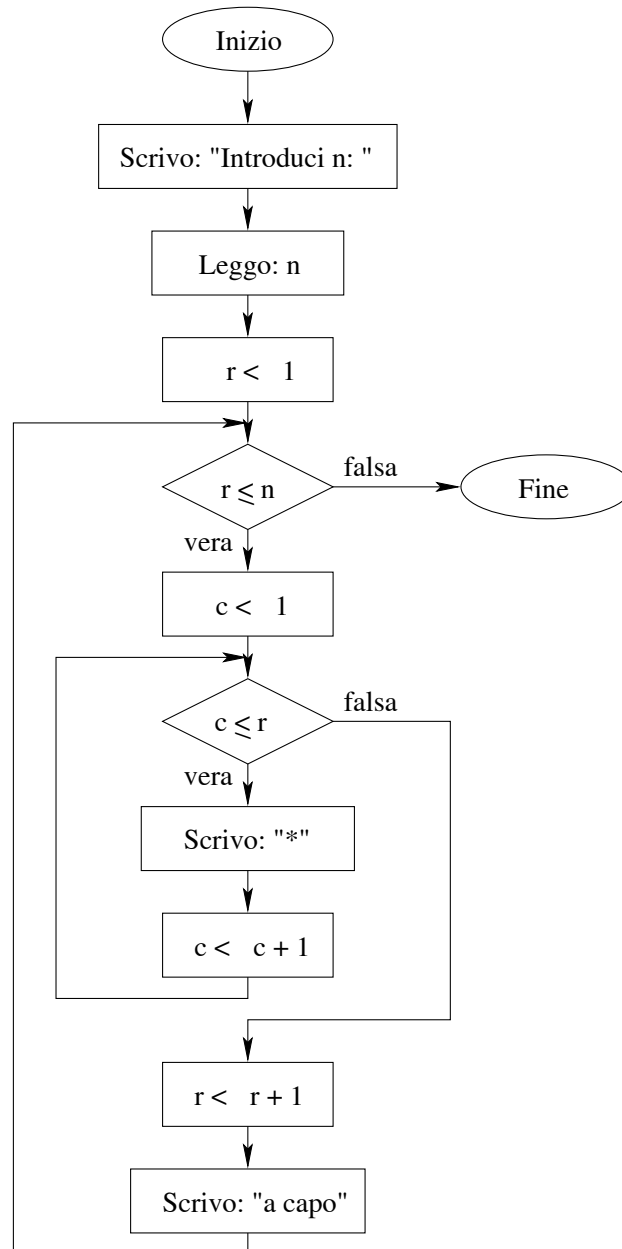


Figura 8: Visualizzazione di un triangolo.

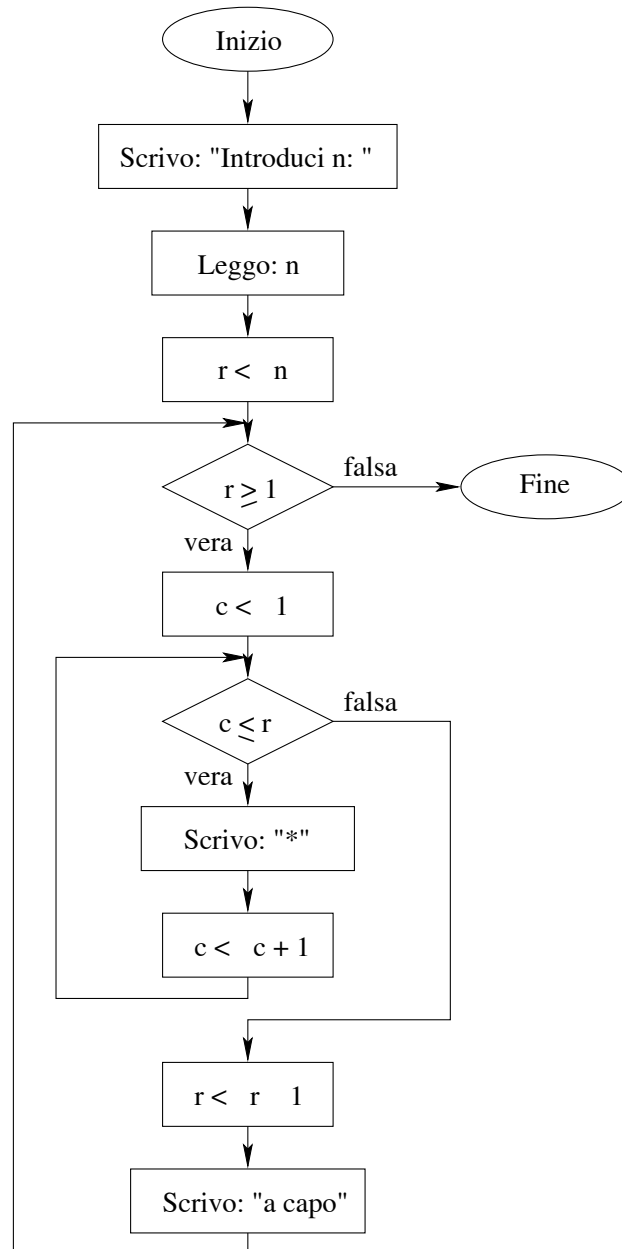


Figura 9: Visualizzazione di un triangolo rovesciato.

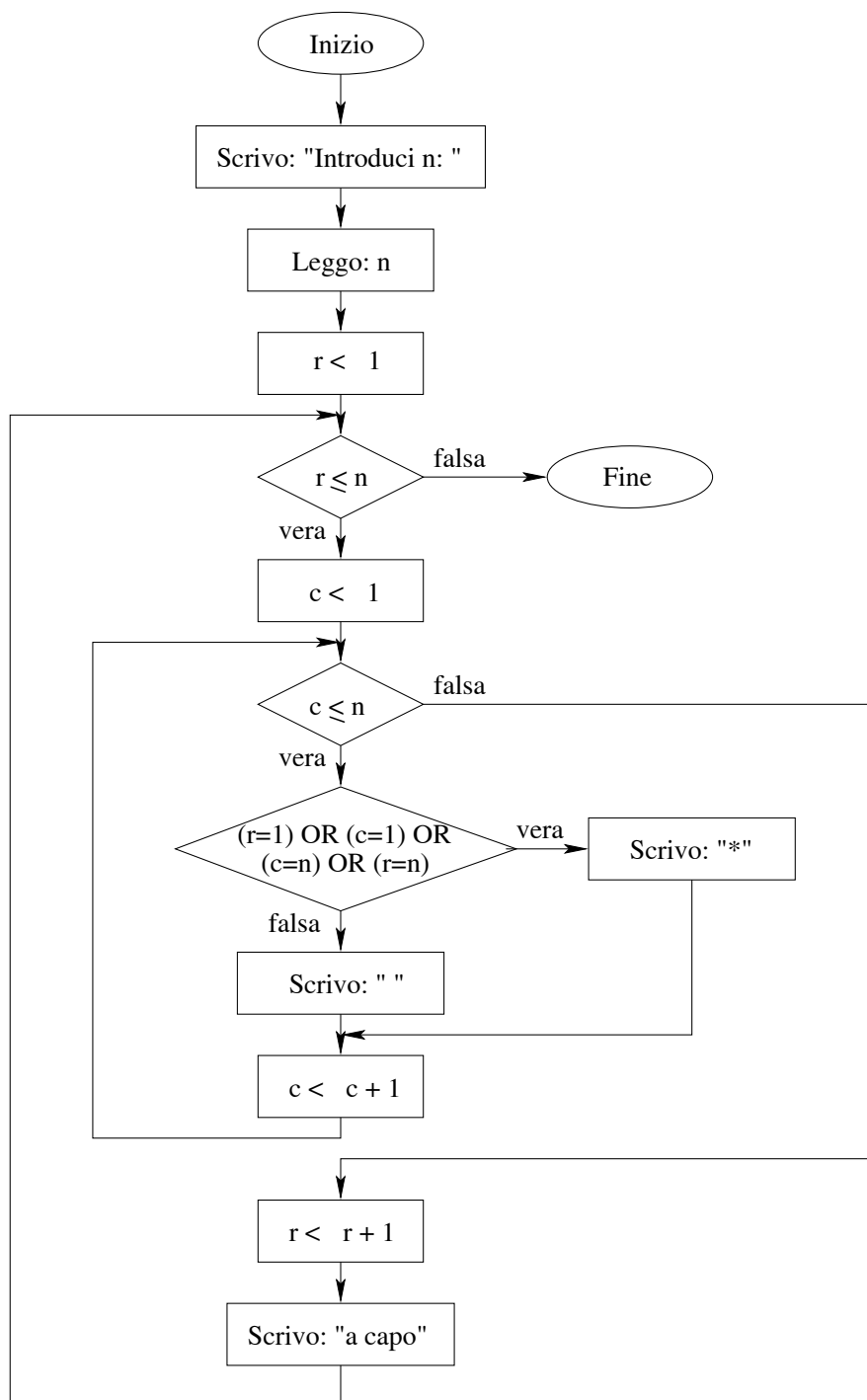


Figura 10: Visualizzazione di un quadrato vuoto.

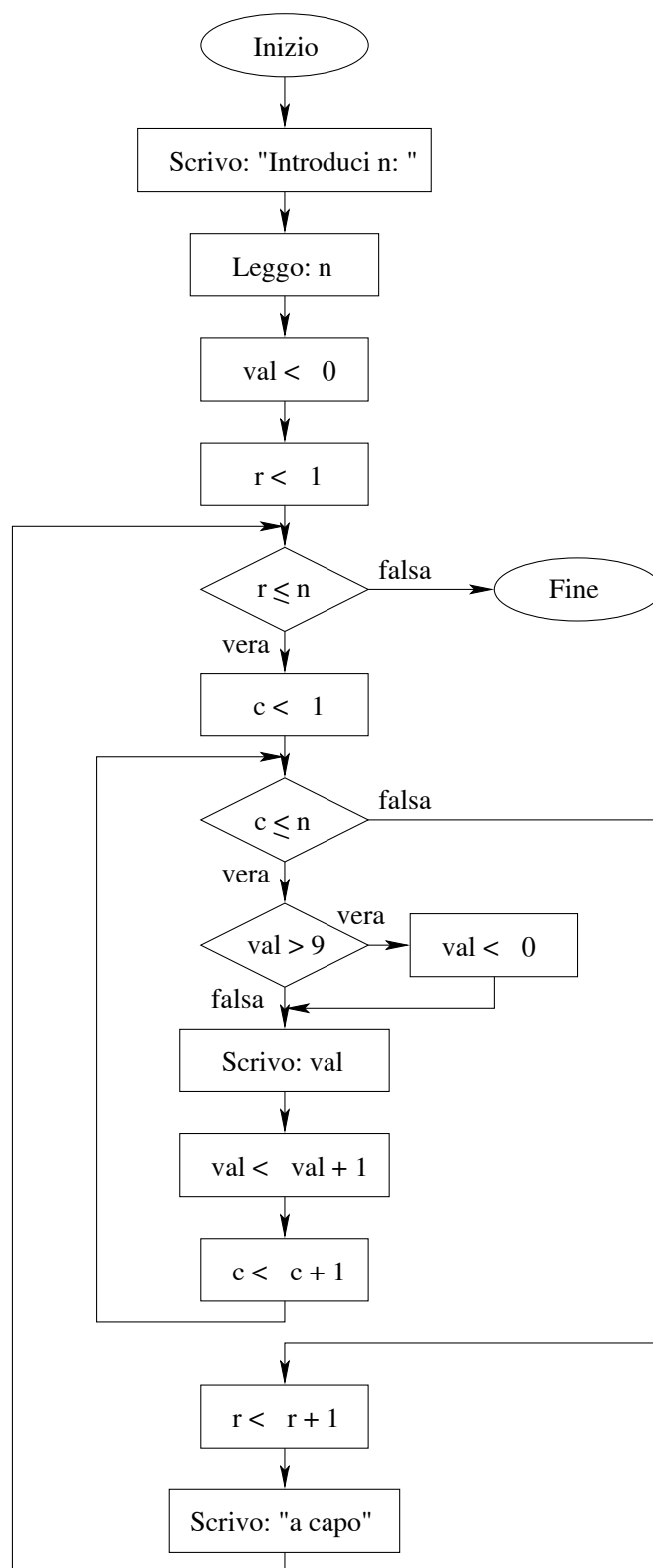


Figura 11: Visualizzazione di un quadrato di cifre.

```
*+++++
-*+++
--*++
---*+
----*
```

Soluzione 1

La soluzione di Figura 12 procede tracciando la figura richiesta seguendo la logica già utilizzata nell'esercizio 5. Il ciclo più esterno effettua un "conteggio" di n righe. Per ciascuna riga il ciclo più interno si occupa di gestire le n colonne. Tuttavia, invece di visualizzare ad ogni iterazione lo stesso simbolo (un asterisco) come nel caso dell'esercizio 5, si stampa un simbolo diverso a seconda della posizione in cui ci si trova rispetto alla diagonale:

- se si è posizionati sulla diagonale, ovvero il valore dell'indice di riga e di colonna coincidono, si visualizza un '*'
- se si è posizionati sopra la diagonale, ovvero il valore dell'indice di riga è minore di quello della colonna, si visualizza un '+'
- se si è posizionati al di sotto della diagonale, ovvero il valore dell'indice di riga è maggiore di quello della colonna, si visualizza un '-'.

Soluzione 2

La soluzione di Figura 13 differisce in maniera sostanziale dalla precedente. Essa si basa sulle considerazioni seguenti. Il quadrato è formato da n righe. Ciascuna riga è costituita da tre sezioni separate:

- una sequenza di simboli '-', eventualmente vuota (nessun simbolo)
- un unico simbolo '*', sempre e comunque presente
- una sequenza di simboli '+', eventualmente vuota (nessun simbolo).

All'interno del ciclo che rappresenta le righe si possono quindi inserire tre sezioni:

- la prima (un ciclo) che visualizza la sequenza di simboli '-'
- la seconda (un'unica operazione di scrittura) che visualizza il simbolo '*'
- la terza (un ulteriore ciclo) che visualizza la sequenza di simboli '+'.

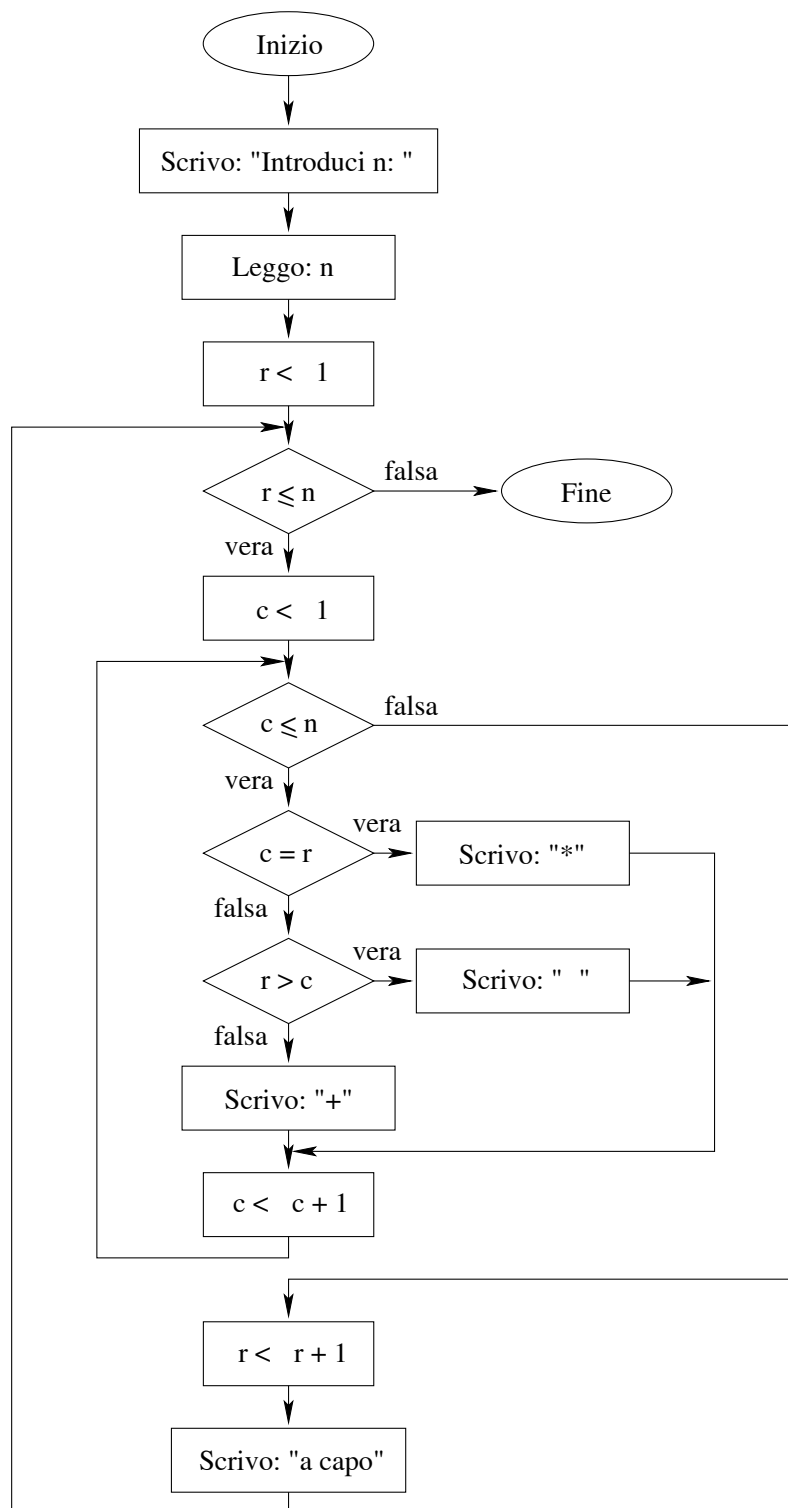


Figura 12: Visualizzazione di un quadrato di simboli: prima soluzione.

Occorre ovviamente ancora determinare quanti simboli ‘-’ e ‘+’ occorre visualizzare sulla generica riga di posizione r (con r che conta le righe e quindi varia da 1 a n compresi). Si può osservare che sulla prima riga si visualizzano 0 simboli ‘-’ ed $(n-1)$ simboli ‘+’, sulla seconda riga si visualizza 1 simbolo ‘-’ e $(n-2)$ simboli ‘+’, etc. Sarà quindi sufficiente realizzare (per ciascuna riga di posizione r) un ciclo che stampi $(r-1)$ simboli ‘-’ e un ciclo che stampi $(n-r)$ simboli ‘+’.

Ovviamente al termine di ciascuna riga occorre ricordarsi di andare a capo.

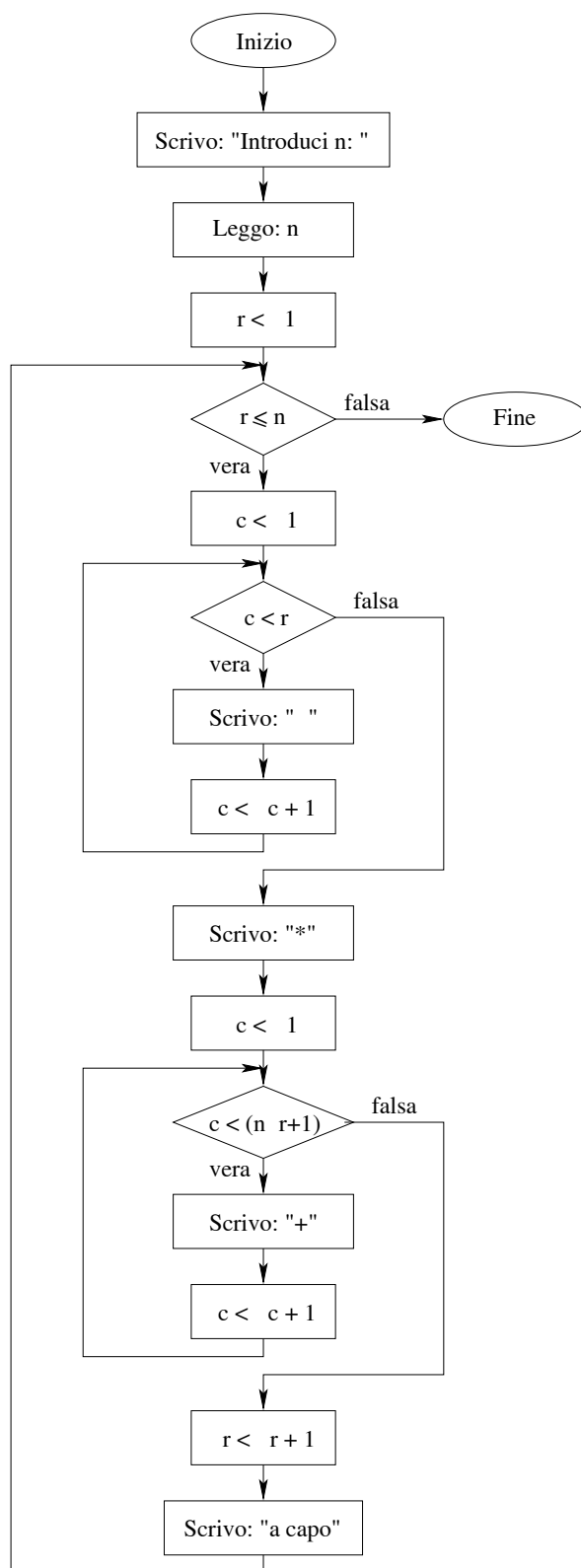


Figura 13: Visualizzazione di un quadrato di simboli: seconda soluzione.