



# Manuale Flowgorithm

Versione 3

Roberto Atzori

« L'informatica non riguarda i computer più  
di quanto l'astronomia riguardi i telescopi. »

*(Edsger Wybe Dijkstra)*

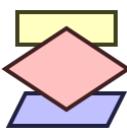
## Sommario

Flowgorithm .....	5
Caratteristiche e funzionalità .....	6
L'interfaccia .....	6
Barra dei pulsanti.....	8
Barra del menù .....	10
Voce di menù “File” .....	10
Stampa.....	11
Voce di menù “Modifica” .....	12
Voce di menù “Aspetto” .....	12
Cambia Lingua .....	13
Dimensionamento .....	13
Scegli la combinazione di colori.....	14
Stile diagramma.....	14
Modifica schema colori .....	15
Modifica Stile Diagramma .....	16
Voce di menù “Programma”.....	18
Velocità di esecuzione .....	18
Rimuovere i punti di arresto.....	19
Nuova Funzione .....	19
Gestione funzioni utente.....	20
Informazioni programma .....	20
Voce di menù “Strumenti” .....	21
Esporta immagine in un file.....	22
Esporta immagine in formato vettoriale .....	22
Voce di menù “Aiuto” .....	23
Le finestre di Flowgorithm.....	24
Finestra “Esecuzione” .....	24
Finestra “Codifica” .....	25
Finestra “Variabili” .....	28
Disposizione finestre .....	28
Finestra grafica “Tartaruga” .....	29
Menù contestuale.....	31
Specifiche.....	32
E ancora.....	32
Il mio primo diagramma di flusso.....	33

Tipi di dato .....	35
Integer (Intero) .....	35
Real (Reale).....	35
String (Stringa).....	35
Boolean (Booleano o Logico).....	35
Identifieri .....	36
Parole chiave .....	36
Operatori .....	36
Precedenze .....	37
Esempi .....	37
Funzioni predefinite .....	38
Matematiche .....	38
Stringa.....	38
Conversione tipo di dati .....	38
Altre .....	39
Costanti predefinite.....	39
Formato File Sorgente .....	40
Esempio .....	40
Istruzioni .....	46
Assegnazione .....	46
Chiamata di Funzione o Procedura .....	47
Commento.....	47
Dichiarazione .....	48
Do .....	48
For.....	49
If.....	50
Input .....	50
Output .....	51
While.....	53
Grafica Tartaruga .....	54
Ruotare .....	54
Avanzare .....	54
Punto iniziale .....	54
Esempi .....	55
Disegno di un fiore.....	55
Scalinate .....	55

Gestione File .....	56
Lettura .....	56
Scrittura .....	56
Chiusura.....	56
Apertura .....	57
Esempi .....	58
Fibonacci con utilizzo di file.....	58
Calcolo dei quadrati.....	58
Interruzioni condizionali .....	59
Per gli Amministratori di Sistema .....	60
Chiavi Registro .....	60
Contatto.....	63
Informazioni.....	63
Licenza .....	63
Ulteriori Informazioni .....	63

# Flowgorithm



Flowgorithm è un'applicazione gratuita che consente di creare programmi utilizzando diagrammi di flusso.

In informatica il **diagramma di flusso** (in inglese detto anche **flowchart**) rappresenta un linguaggio di modellazione grafico per rappresentare il flusso di controllo ed esecuzione di algoritmi, procedure, istruzioni operative (in senso lato).

Esso consente di descrivere in modo schematico ovvero grafico:

- le operazioni da compiere, rappresentate mediante sagome convenzionali (rettangoli, rombi, esagoni, parallelogrammi, rettangoli smussati...), ciascuna con un preciso significato logico e all'interno delle quali un'indicazione testuale descrive tipicamente l'attività da svolgere
- la sequenza nella quale devono essere compiute sono rappresentate con frecce di collegamento.

Per questa loro connotazione topologica i diagrammi di flusso possono essere ricondotti alla classe più ampia dei diagrammi a blocchi, che a loro volta rientrano nell'ancora più vasta categoria delle mappe concettuali, utilizzate soprattutto per la descrizione e la rappresentazione delle informazioni e della conoscenza.

In genere, i programmi sono scritti utilizzando un editor di testo. A seconda del linguaggio di programmazione, questo può risultare facile o molto difficile per un programmatore alle prime armi. Molti linguaggi richiedono la stesura di diverse righe di codice solo per arrivare alla visualizzazione di una semplice frase come "Ciao, mondo!".

Tuttavia, utilizzando Flowgorithm, è possibile utilizzare dei simboli per rappresentare le diverse azioni che si desidera che il programma esegua. Quindi, ci si può concentrare su un algoritmo (**Pensiero Computazionale**) piuttosto che su tutte le sfumature e i dettagli propri di un tipico linguaggio di programmazione (**Coding**).

È possibile eseguire i programmi direttamente in Flowgorithm. Ma, se si vuole imparare un linguaggio di programmazione ad alto livello, Flowgorithm può convertire il diagramma di flusso in molti linguaggi popolari. Questi includono: C#, C++, Delphi/Pascal, Java, JavaScript, Lua, Perl, Python, QBasic, Ruby, Swift 2, Visual Basic .NET e Visual Basic for Applications (utilizzato in Microsoft Office), TypeScript, Scala.

Una importante collezione di algoritmi di vario genere, famosi o meno, creati con il software Flowgorithm, possono essere visionati e/o scaricati dal mio eBook presente nel mio sito all'indirizzo <http://robatz.altervista.org/blog/flowcharts-book/>.

# Caratteristiche e funzionalità

## L'interfaccia

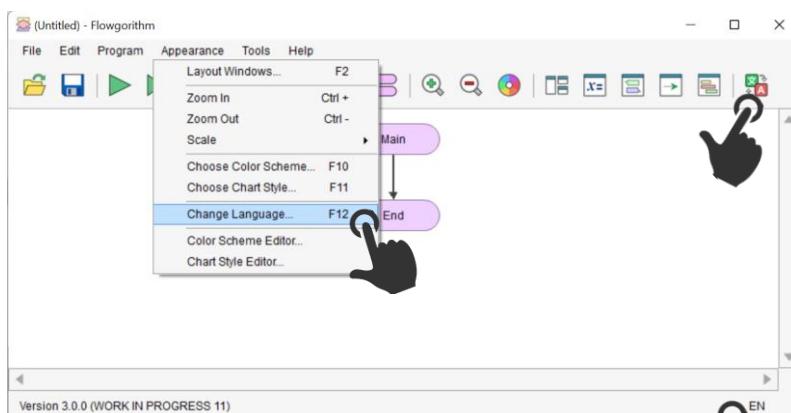
Flowgorithm si presenta con una interfaccia molto semplice ed intuitiva. La zona menù è corredata di tutte le funzionalità fornite dal software. La zona pulsanti è munita delle principali funzionalità per una facile implementazione dell'algoritmo, una facile esecuzione e un facile controllo dello stesso.



## Supporto Multilingua

Flowgorithm è tradotto in molti linguaggi:

- Cinese (Simplificato)
- Ceco
- Inglese - U.S. e British
- Francese
- Galiziano
- Tedesco
- Ungaro
- Italiano
- Giapponese
- Portoguese
- Spagnolo – Messicano e Castigliano
- ....



Per cambiare linguaggio (inizialmente) possiamo cliccare la voce di menù **Tools** → **Change Language**

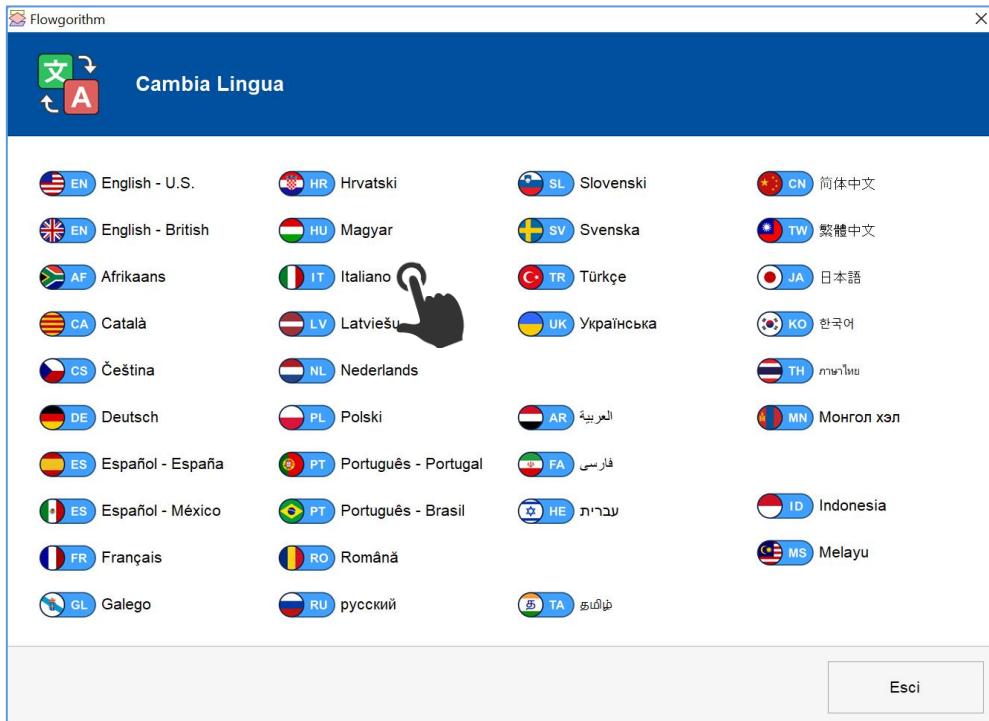


Oppure fare doppio click col tasto sinistro del mouse in basso a destra della finestra in corrispondenza della figura

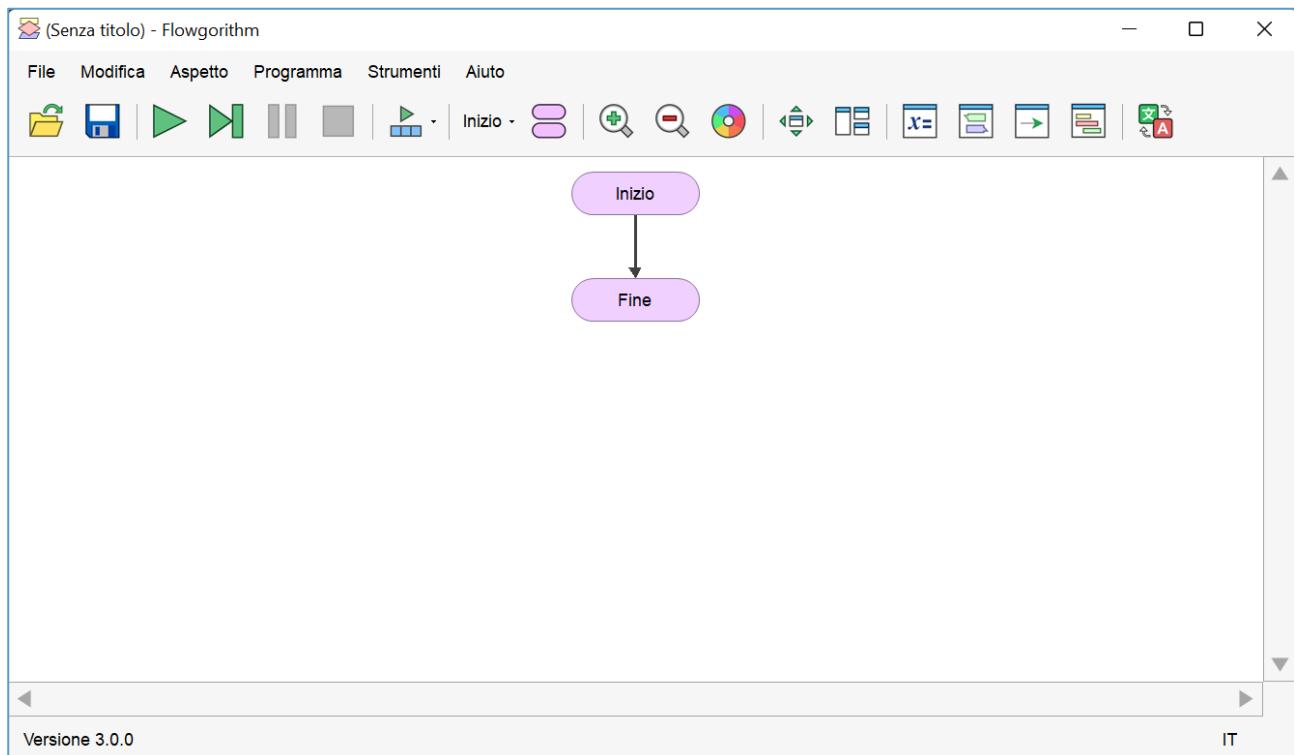
o nella icona nel menu



Verrà aperta la finestra di selezione nuovo linguaggio:



Selezionare la voce “Italiano” e l’intera interfaccia di Flowgorithm userà l’idioma italiano.



e il simbolo cambierà in ad indicare la lingua correntemente in uso.

## Barra dei pulsanti

Immediatamente sotto la barra dei menu è presente la **barra degli strumenti standard**, che contiene i pulsanti dei comandi utilizzati più di frequente (tasti di scelta rapida) per la gestione dei file e per l'esecuzione delle operazioni più importanti.



Pulsante	Descrizione
	Carica un flowchart precedentemente creato e salvato
	Salva il flowchart
	Esegui interprete flowchart in automatico secondo la velocità impostata
	Passo successivo modalità utente
	Pausa nella modalità automatica
	Interrompi interprete flowchart
	Imposta velocità interprete in esecuzione automatica
	Selettore programma principale / funzioni
	Gestione funzioni utente
	Aumenta ingrandimento
	Diminuisci ingrandimento
	Combinazioni colori
	Auto Layout (la finestra verrà posizionata in primo piano)
	Disposizione e selezione finestre
	Finestra 'Variabili'
	Finestra 'Esecuzione'
	Finestra grafica Tartaruga
	Finestra 'Codifica'
	Selezione lingua

Attenzione: i pulsanti della barra degli strumenti standard corrispondono alle voci più utilizzate presenti nella barra dei menu; la barra degli strumenti standard, perciò, ha solo la funzione di rendere tali voci più accessibili. Per esempio, se si vuole salvare il documento su cui si sta scrivendo è sufficiente utilizzare l'apposito pulsante presente sulla barra degli strumenti senza dover ogni volta aprire il menu **File** e selezionare la voce **Salva**.

Per selezionare i pulsanti della barra degli strumenti standard tramite tastiera è possibile:

- posizionarsi sulla barra dei menu tramite il tasto **Alt**;
- utilizzare la combinazione di tasti **Control** più **Tab**.

In questo modo viene selezionato il primo pulsante della barra; per spostarsi fra i pulsanti è sufficiente utilizzare il tasto **Tab** per muoversi da sinistra verso destra e la combinazione di tasti **Shift** più **Tab** per muoversi da destra verso sinistra. Una volta individuato il pulsante d'interesse occorre premere il pulsante **Invio** per attivarlo.

## Barra del menù

La **barra dei menu** presenta la maggior parte dei comandi disponibili nel programma, suddivisi per menu (elenchi di scelte). Tramite tastiera è possibile consultare la barra dei menu premendo il tasto **Alt** e spostandosi con la **Freccia Destra** o **Sinistra** per leggere le intestazioni di ogni menu. Una volta individuato quello di interesse, è possibile leggere le voci contenute al suo interno utilizzando i comandi **Freccia Giù** o **Su**. Per attivare una voce è sufficiente, dopo averla individuata, premere il tasto **Invio**.

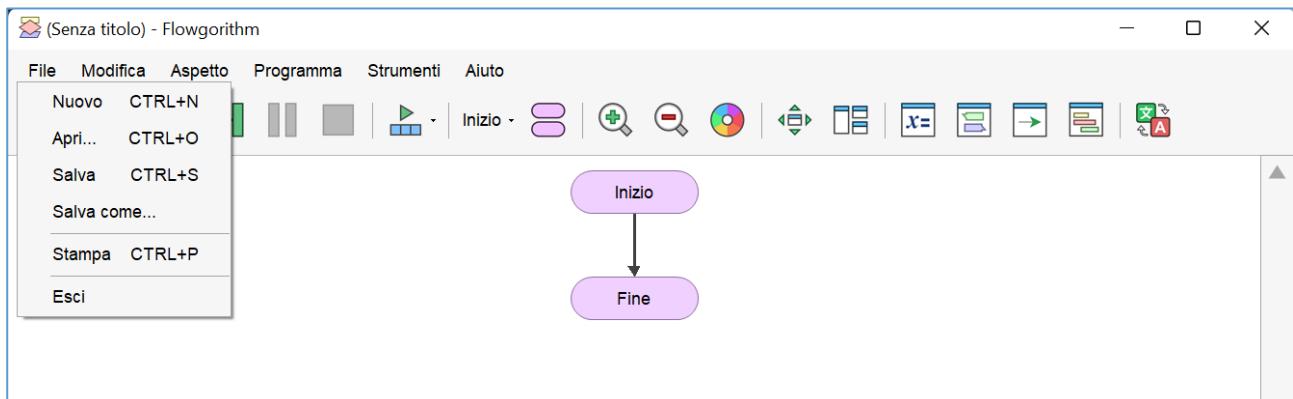
Per aprire ciascuno dei menu disponibili, si possono utilizzare anche le seguenti combinazioni di tasti, dette anche tasti caldi (hot):

- **Alt più F** per il menu **File**;
- **Alt più M** per il menu **Modifica**;
- **Alt più P** per il menu **Programma**;
- **Alt più S** per il menu **Strumenti**;
- **Alt più A** per il menu **Aspetto**.

Per chiudere un menu è sufficiente selezionare un comando del menu, attivare un altro menu o premere il tasto **Alt**.



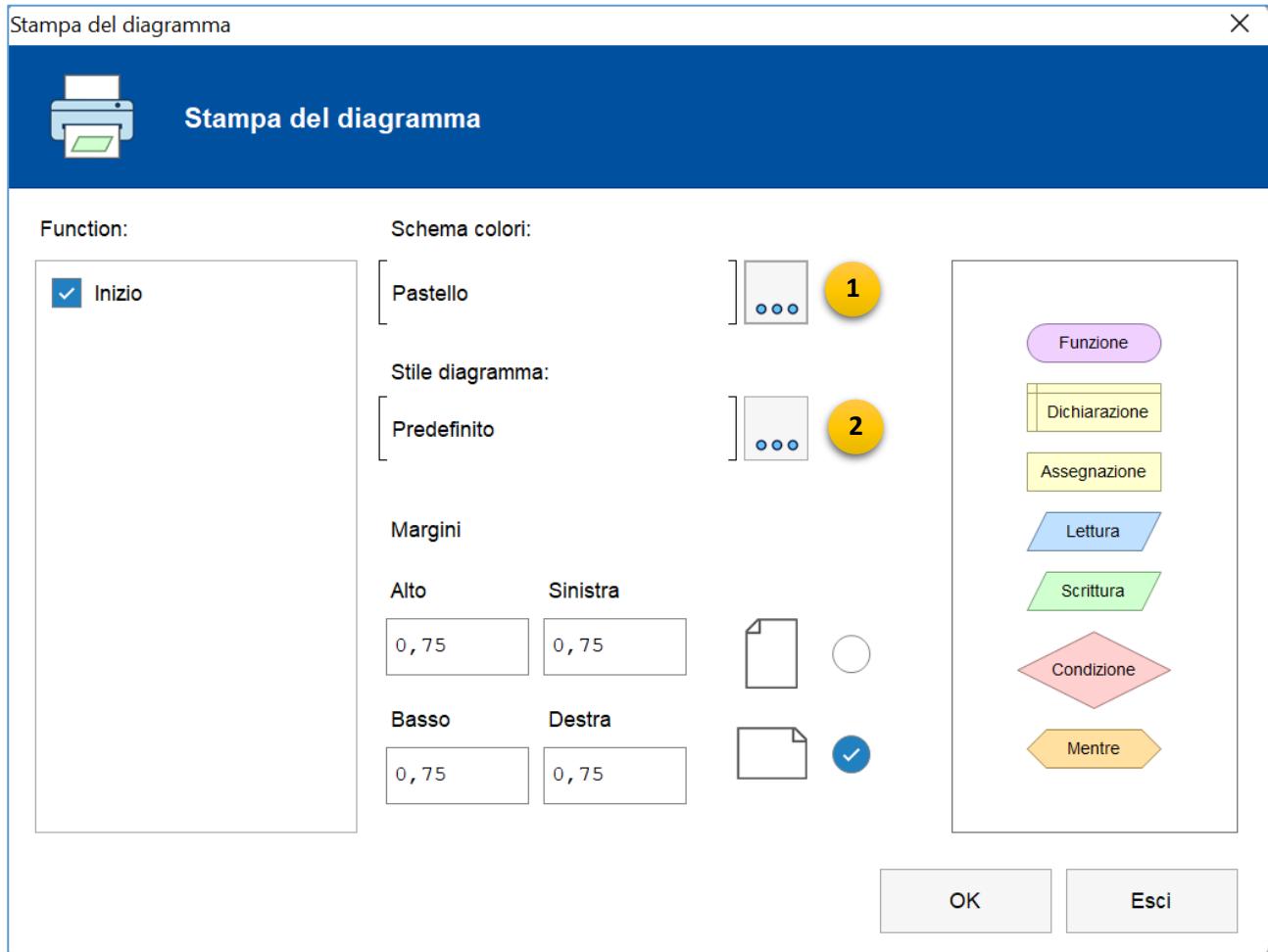
Voce di menù “File”



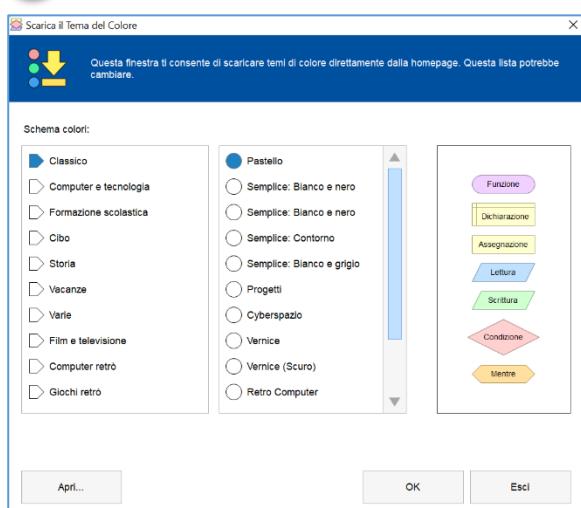
Voce menù	Descrizione
<b>Nuovo</b>	Crea un nuovo flowchart
<b>Apri</b>	Carica un flowchart precedentemente creato e salvato
<b>Salva</b>	Salva il flowchart
<b>Salva come...</b>	Salva il flowchart con la possibilità di cambiare nome allo stesso e cartella dove salvarlo
<b>Stampa</b>	Per stampare la zona correntemente visualizzata
<b>Esci</b>	Uscire dal programma

## Stampa

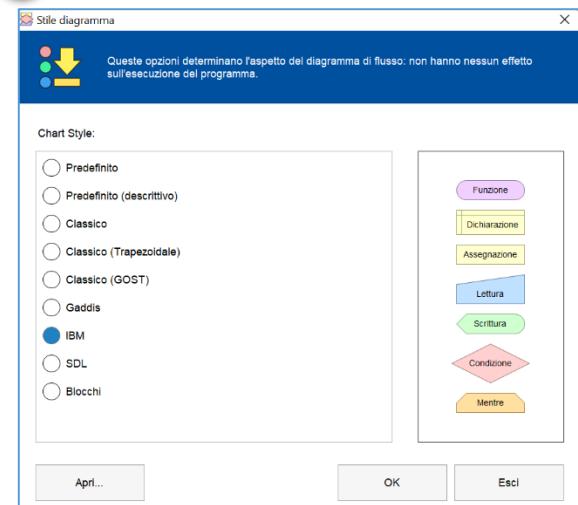
Selezionando la voce di menù “Stampa” si aprirà la seguente finestra:



1

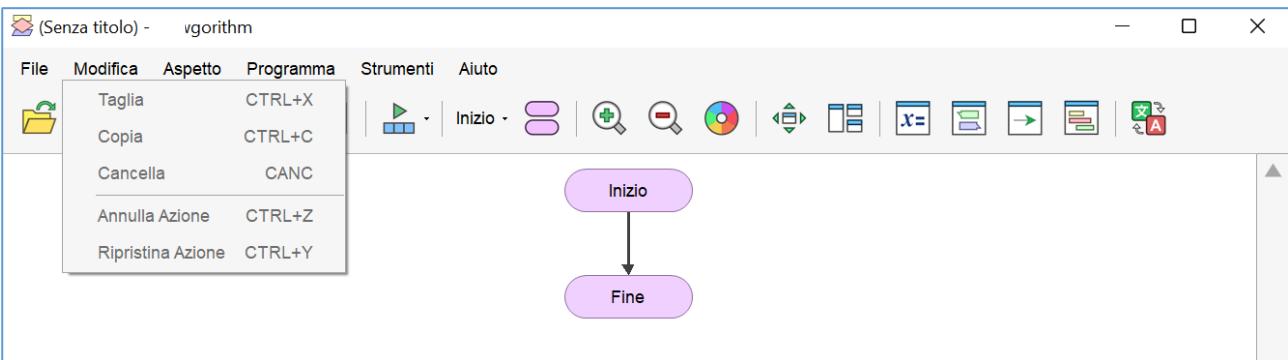


2



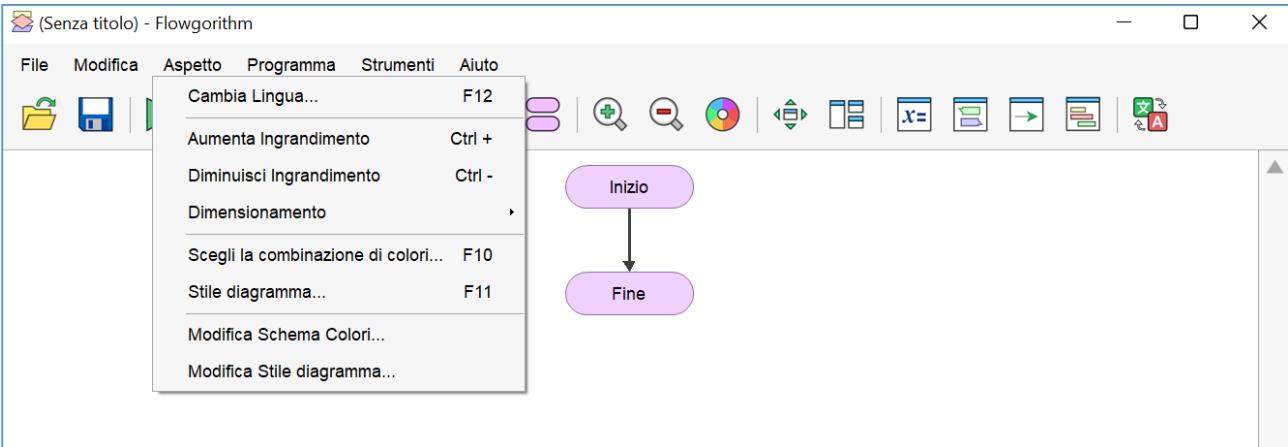
attraverso la quale è possibile inserire delle impostazioni personalizzate per la stampa.

### Voce di menù “Modifica”



Voce menù	Descrizione
<b>Taglia</b>	Cancellate l'elemento selezionato copiandolo in memoria
<b>Copia</b>	Copia in memoria l'elemento selezionato
<b>Cancella</b>	Cancella l'elemento selezionato
<b>Annulla Azione</b>	Annulla operazione inserimento forme, digitazione, ...
<b>Ripristina Azione</b>	Ripristina operazione annullata

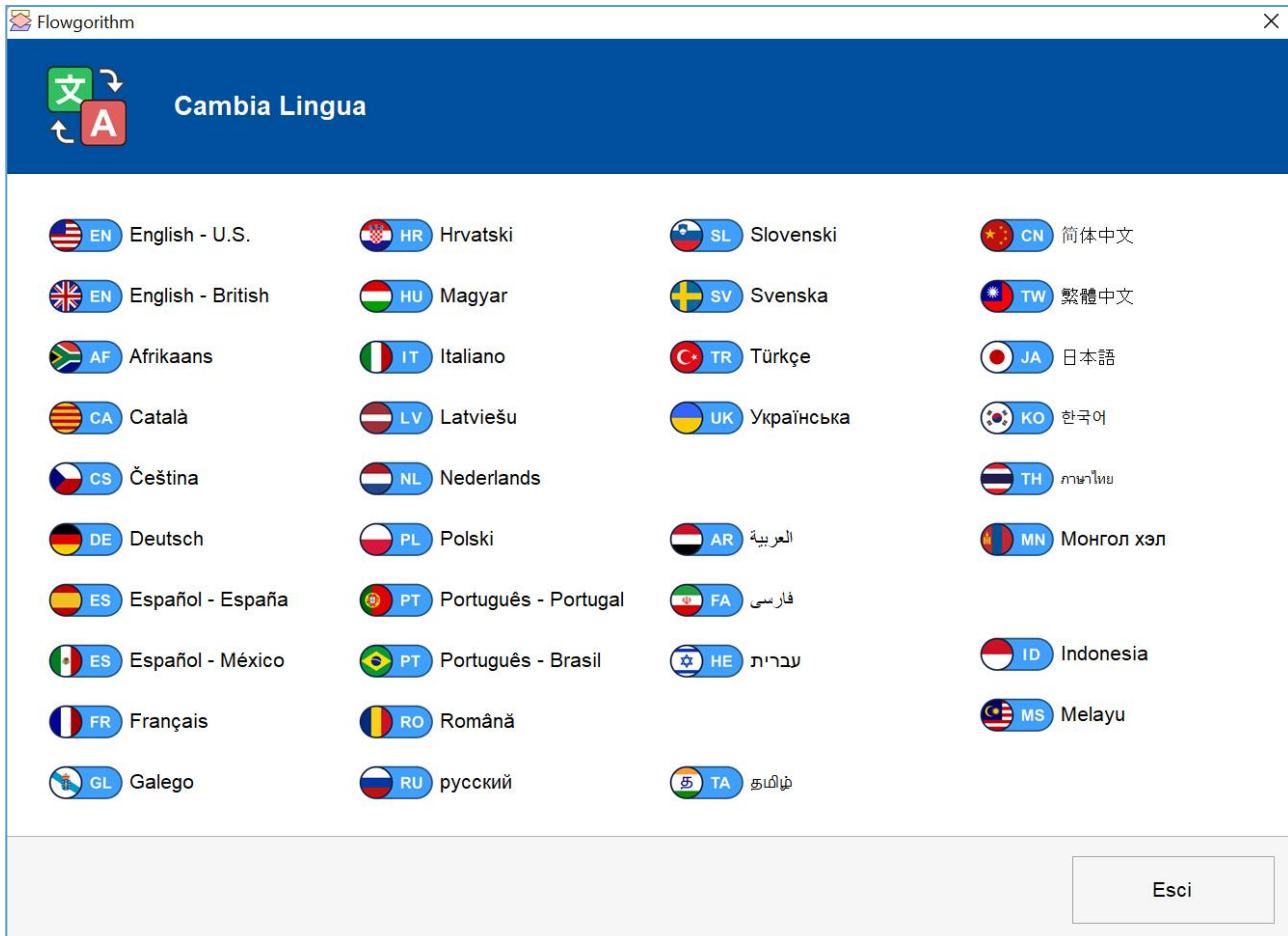
### Voce di menù “Aspetto”



Voce menù	Descrizione
<b>Cambia Lingua</b>	Permette di cambiare l'idioma
<b>Ripristina Azione</b>	Ripristina operazione annullata
<b>Aumenta ingrandimento</b>	Ingrandisce la zona del flowchart
<b>Diminuisci ingrandimento</b>	Rimpicciolisce la zona del flowchart
<b>Dimensionamento</b>	Ridimensiona intera finestra Flowgorithm
<b>Scegli la combinazione di colori</b>	Selezionare la combinazione di colori
<b>Stile diagramma ...</b>	Seleziona lo stile del diagramma

<b>Modifica Schema Colori ...</b>	Modifica colorazione temi
<b>Modifica Stile diagramma ...</b>	Modifica lo stile del diagramma

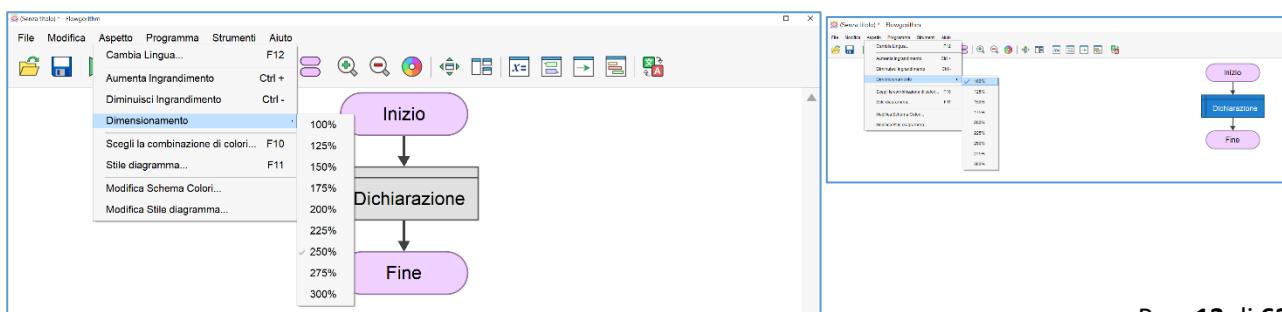
### Cambia Lingua



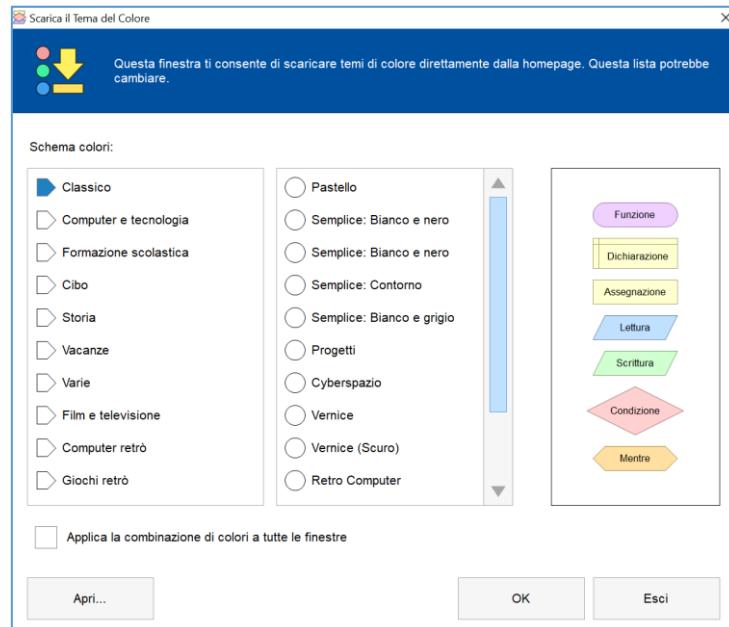
### Dimensionamento

L'applicazione ora è "DPI-aware". Ciò significa che la finestra si "ridimensionerà" su schermi più grandi (con grande risoluzione) utilizzando un fattore di scala. Normalmente questo è impostato al 125% o 150%. L'utente può anche modificare il fattore di scala corrente dal menu Strumenti → Dimensionamento. Prima di questo cambiamento, Windows avrebbe semplicemente, su schermi più grandi come ad esempio un 4K, rimpicciolito tutta la grafica rendendola poco leggibile.

Questa possibilità è utile ad esempio in ambito laboratoriale nel momento in cui si utilizza una risoluzione molto dettagliata (ad esempio 4k) di difficile lettura agli ultimi posti nel laboratorio.

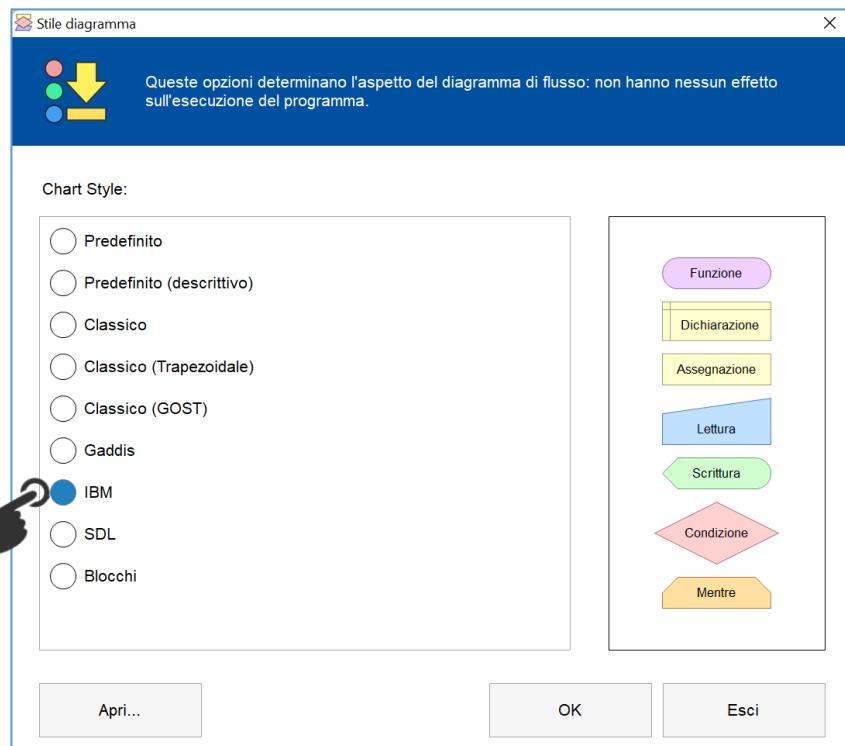


### Scegli la combinazione di colori



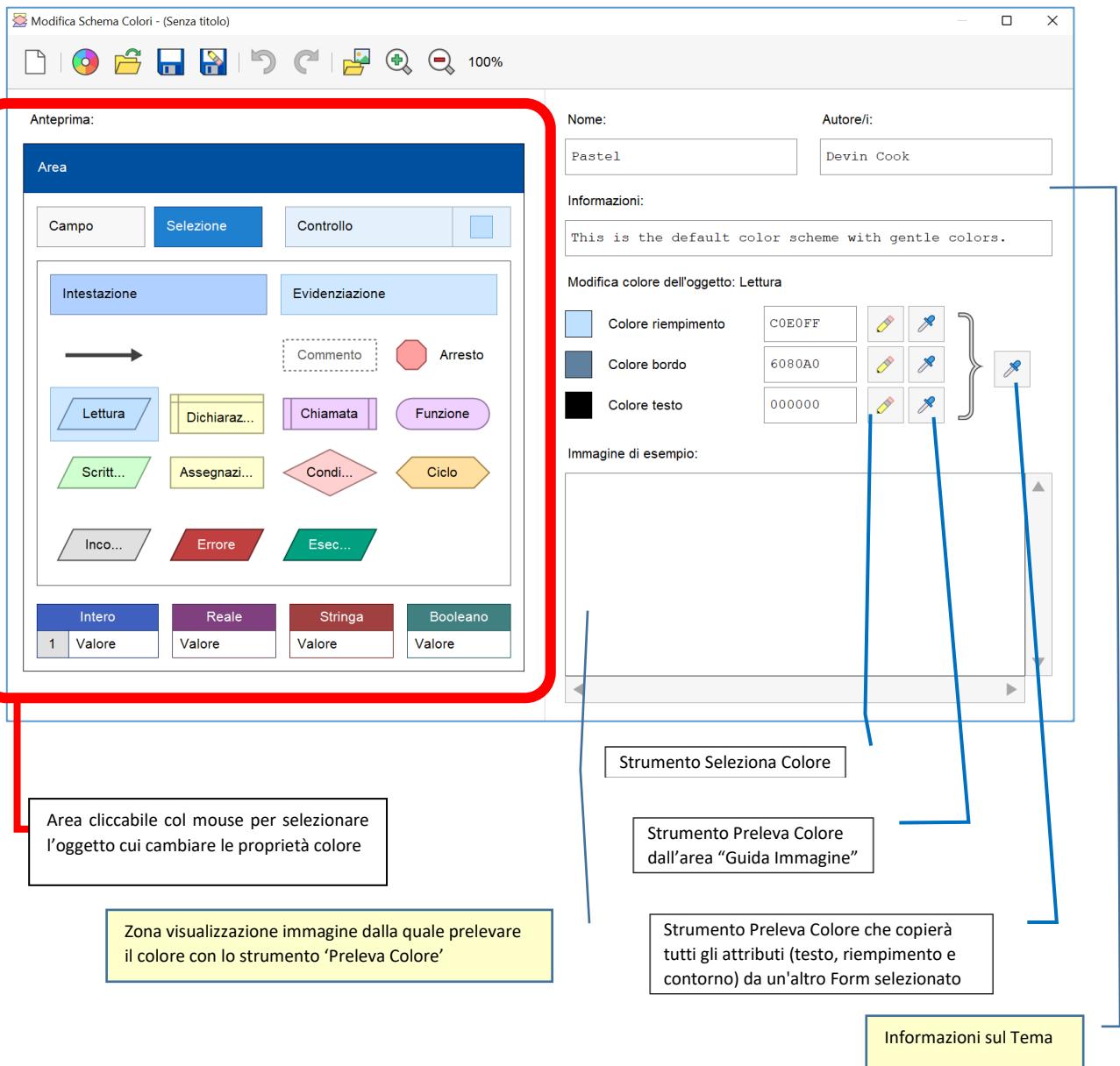
### Stile diagramma

Particolare importanza riveste la scelta della selezione “Stile del diagramma” che porta all’apertura di una nuova finestra:



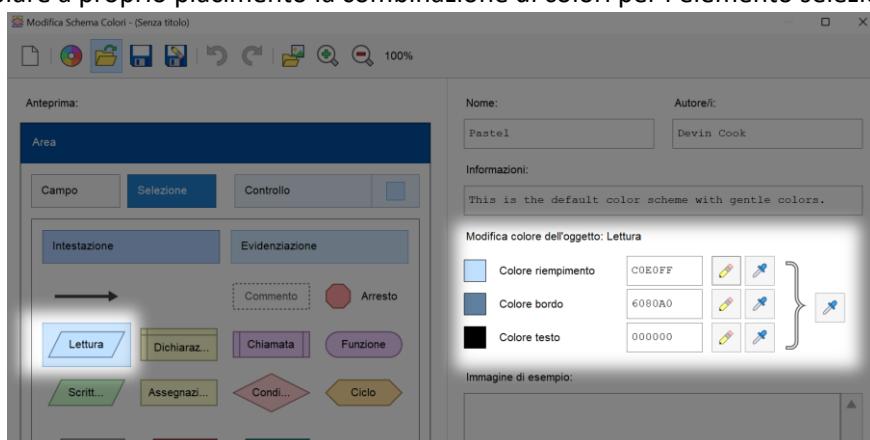
## Modifica schema colori

Selezionando questa voce, è possibile modificare la grafica dei simboli predefiniti

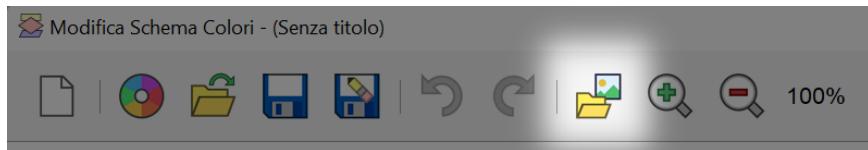


Questo permetterà di creare/modificare qualsiasi stile del diagramma di flusso.

Cliccando su qualsiasi elemento della zona “Anteprima” (a sinistra), si avrà la possibilità (sulla destra) di cambiare a proprio piacimento la combinazione di colori per l’elemento selezionato.



o, addirittura, colore di un elemento grafico importandolo dal proprio PC.



### Modifica Stile Diagramma

E' possibile personalizzare ogni elemento selezionato nella zona "Anteprima" a sinistra, assegnandoci una qualsiasi forma tra quelle presenti nella zona "Forme" a destra.

All'indirizzo [Flowgorithm](#) - [Download Color Schemes](#) si possono trovare gli schemi preinstallati ed altri nuovi, la cui colorazione può essere modificata dall'utente.



**Color Schemes**

**Seasonal**

- Christmas (The colors of a snowy day)
- Christmas Tree
- Easter
- Fall
- Halloween (The colors of the creepy wallpaper)

**Science Fiction**

- Doctor Who (The colors of the TARDIS)
- LCARS (The colors from Star Trek: The Next Generation)
- The Orville (The new Star Trek inspired show)
- Mad Science (Mystery Science Theater 3000 inspired)

**Movies & Television**

- Hoplite (The movie "300")
- Harrison's Dynasties (The color of the clothes from the classic film)
- Robbie Rotten (Lacy Town's greatest villain. We are #1)
- Tea Party (Alice in Wonderland, happy un-birthday!)
- Mr. Meeseeks (Colors based on the popular character from Rick and Morty)

**Harry Potter**

These color schemes play homage to the fantastic Harry Potter books by J.K. Rowling.

- Gryffindor
- Hufflepuff
- Ravenclaw
- Slytherin

**Computer Science**

- DOS (A classic color scheme used by computer programmers.)
- Solarized - Outline (Outline version of the scheme)
- Windows 8 (The pleasant colors from the much-maligned Start Screen.)
- Windows 8 - Border (The same scheme, but the shapes have a black border for contrast.)

**Computer Hardware**

- Circuit Board
- HP Calculator (The classic orange, blue and brown.)
- Liquid Crystal
- Red LED (For that creepy electronics look.)
- Terminal (The classic green computer screen.)

**Classic Video Games**

- Bard's Tale III (A classic RPG game on the Commodore 64)
- Battle Chess (The colors of the classic 1980's game.)
- Super Mario Brothers (Nintendo Entertainment System (NES))
- Super Mario Brothers - Castle Level (Nintendo Entertainment System (NES))
- Super Mario Brothers 3 (Nintendo Entertainment System (NES))

**Classic Computer Systems**

- Commodore Amiga Workbench (Colors of a very early GUI)
- Commodore 64
- CGA - Palette 1 - Low (Black, cyan, magenta, gray)
- CGA - Palette 1 - High (Black, light cyan, light magenta, white)
- CGA - Palette 2 - Low (Black, cyan, magenta, white)
- CGA - Palette 2 - High (Black, light green, pink, yellow)
- GEOS (An early GUI on the Commodore 64.)
- Windows 3.1 Hotdog Stand (Yes, this was real.)

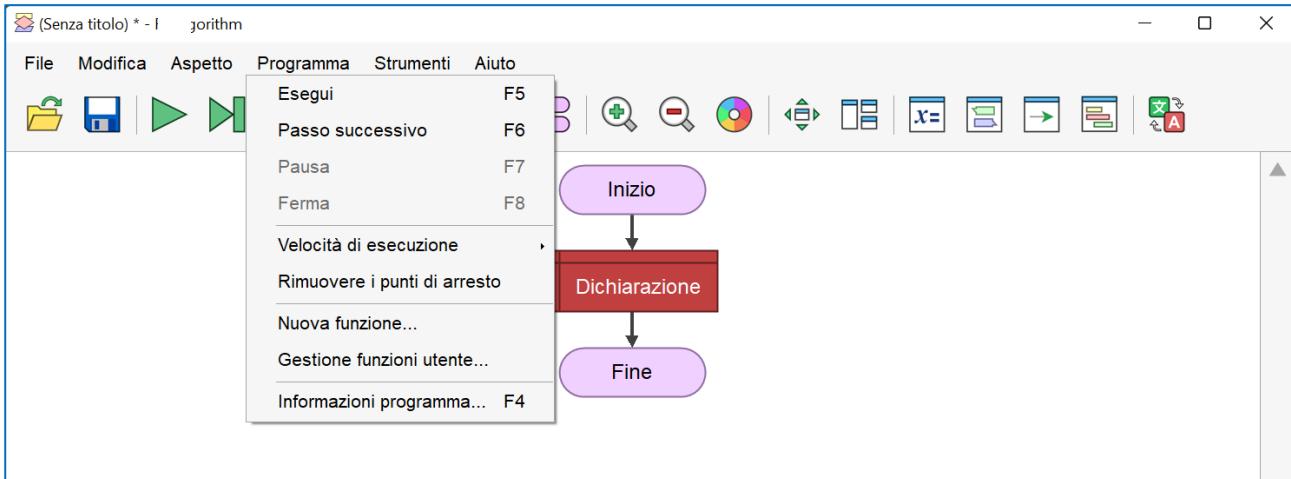
**Miscellaneous**

- 1970's - Brown
- 1970's - Green
- Chalkboard - Black
- Chalkboard - Green
- City Lights
- Flag of California
- Medical Clinic
- Traffic - At Night (This was the original traffic color scheme.)

**Created A Scheme?**

If you created a color scheme, and would like to see it on the site, please send it to Devin @ Flowgorithm.org.

## Voce di menù "Programma"



Voce menù	Descrizione
<b>Esegui</b>	Esegue il flowchart (interpretazione)
<b>Passo successivo</b>	Esegue passo passo il flowchart
<b>Pausa</b>	Pausa esecuzione
<b>Ferma</b>	Interrompe l'esecuzione
<b>Velocità di esecuzione</b>	Selezione della velocità di esecuzione in modalità automatica
<b>Rimuovere i punti di arresto</b>	Rimuove tutti i punti di interruzione inseriti
<b>Nuova funzione</b>	Aggiunge una nuova funzione
<b>Gestione funzioni utente ...</b>	Gestione delle funzioni
<b>Informazioni programma ...</b>	Proprietà del programma

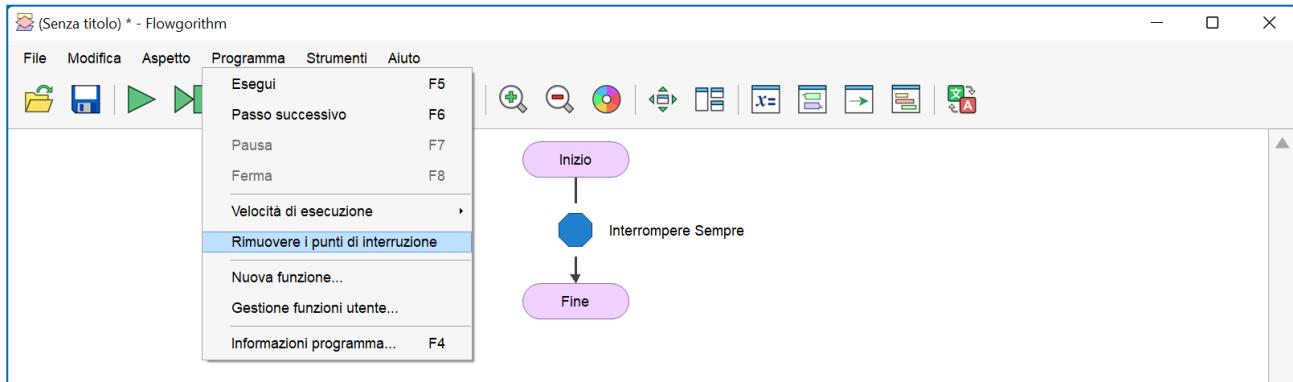
### *Velocità di esecuzione*

Selezionando la voce di menù **Velocità di esecuzione** si aprirà un altro menù attraverso il quale sarà possibile individuare la velocità di esecuzione in automatico delle istruzioni.



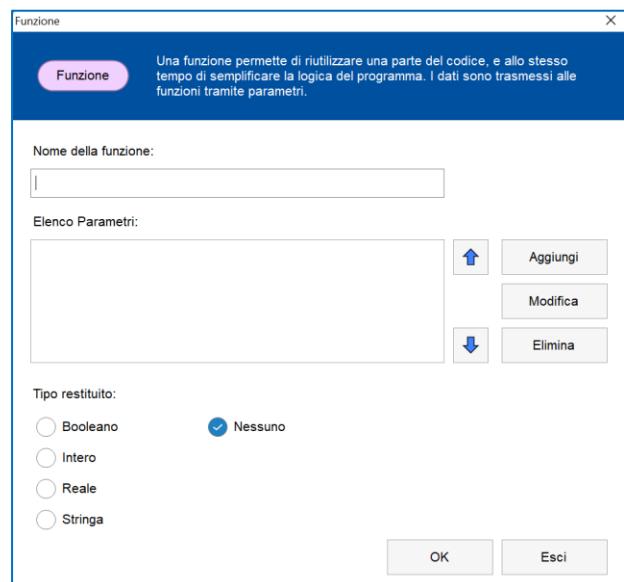
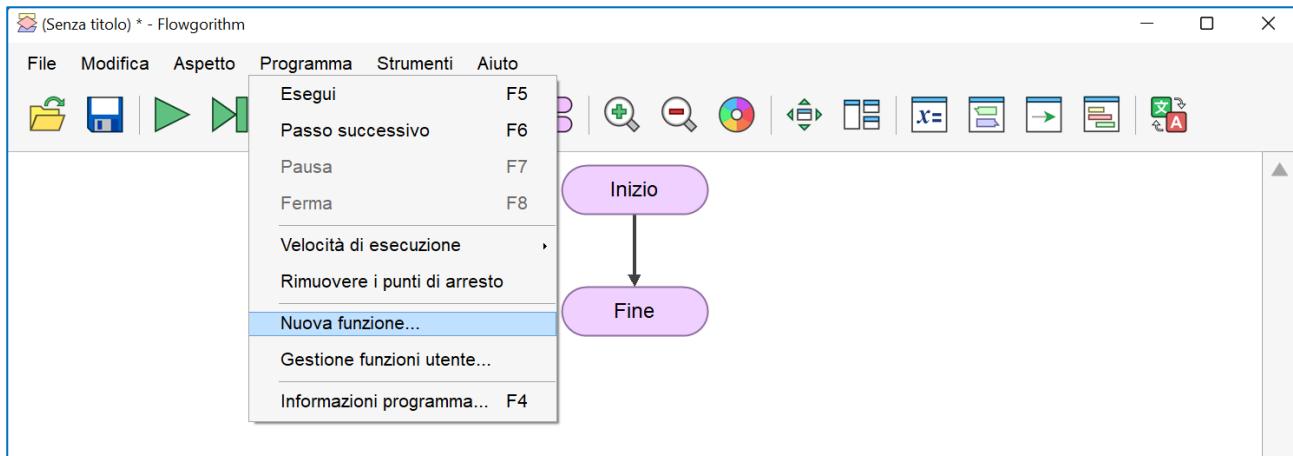
## Rimuovere i punti di arresto

Questa voce permette di rimuovere tutti i punti di arresto nel flowchart

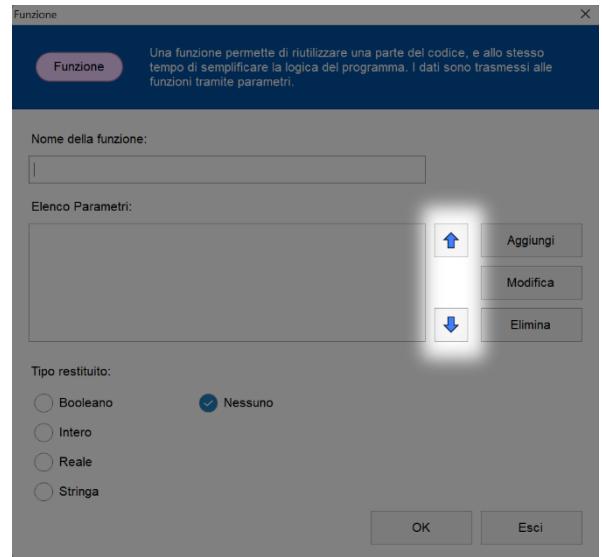


## Nuova Funzione

Selezionando la voce di menù “Nuova funzione” si aprirà un’altra finestra tramite la quale sarà possibile creare procedure e funzioni accessori e utili al funzionamento dell’algoritmo stesso.

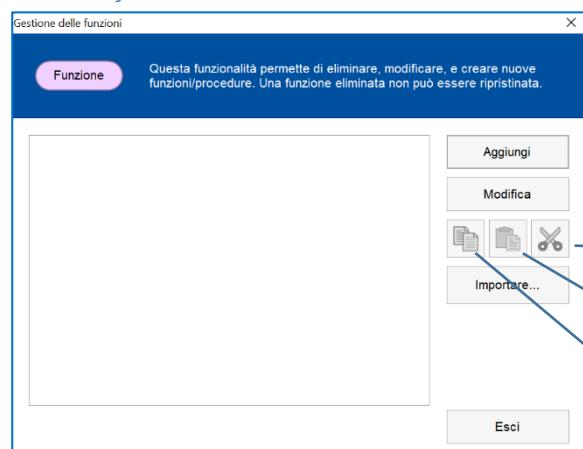


Ovviamente questa zona necessita di conoscenze di programmazione avanzata per poterla sfruttare appieno.



I pulsanti freccia servono per modificare l'ordine dei parametri della funzione. Modificando l'ordine dei parametri si deve prestare attenzione a quanto scritto nel flowchart come codice!!

#### Gestione funzioni utente

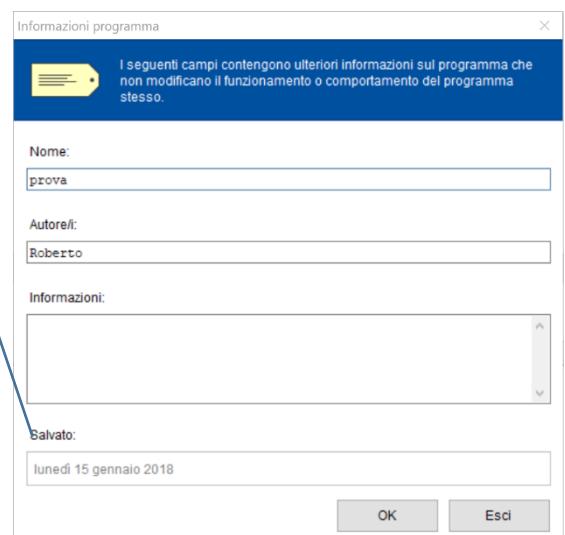
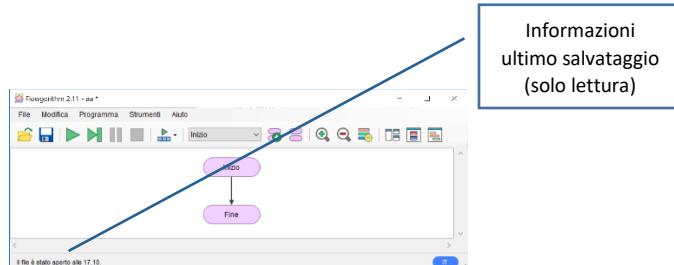


Selezionando la voce di menù **Gestione funzioni utente** si accede alla lista delle funzioni create per l'algoritmo corrente e tramite la quale è possibile modificare le funzioni create, duplicare (copia e incolla) o cancellare quelle che non servono più.

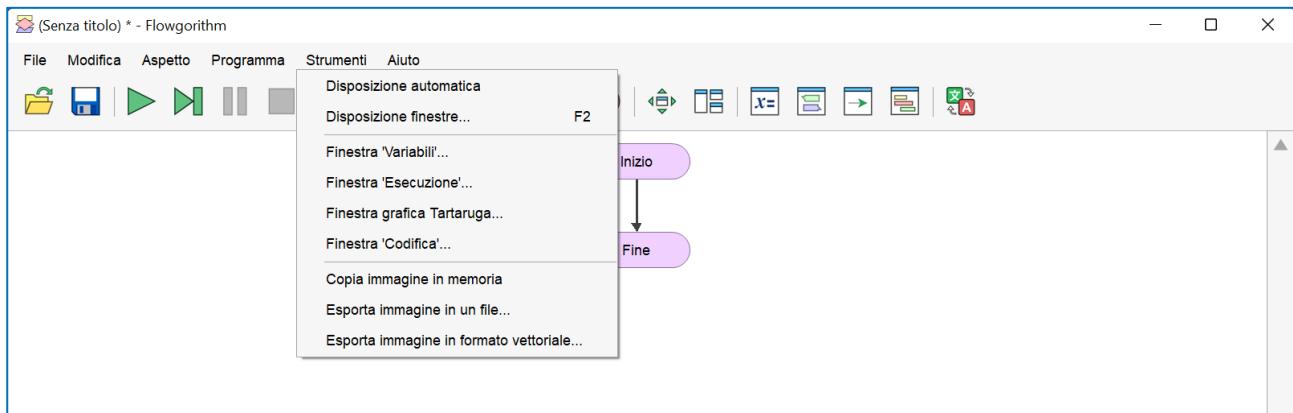
#### Informazioni programma

Selezionando la voce di menù **Informazioni programma** (oppure cliccando sul tasto funzione **F4**) viene attivata un'altra finestra che offre la possibilità di poter inserire ulteriori informazioni sul flowchart.

**Queste informazioni, comunque, non modificano il funzionamento del flowchart.**



## Voce di menù “Strumenti”



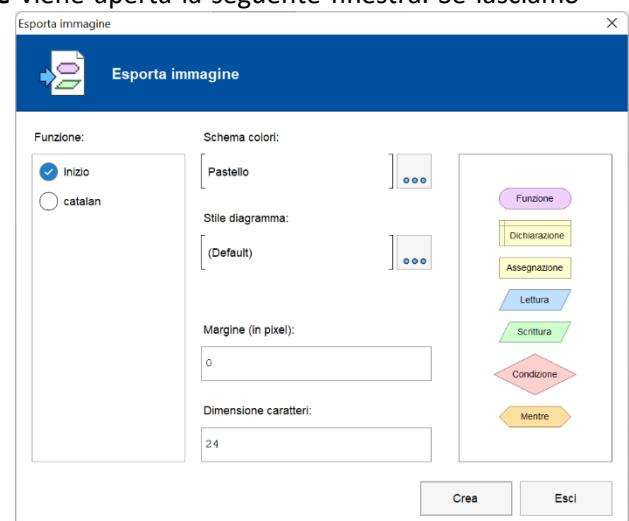
Voce menù	Descrizione
<b>Disposizione automatica</b>	Disposizione automatica delle finestre di Flowgorithm
<b>Disposizione finestre ...</b>	Disposizione delle finestre nel desktop
<b>Finestra 'Variabili' ...</b>	Abilita la finestra per il controllo delle variabili utilizzate nel programma
<b>Finestra 'Esecuzione' ...</b>	Abilita la finestra per monitorare l'esecuzione del programma e per inserire i valori da tastiera
<b>Finestra grafica Tartaruga</b>	Utilizzo della programmazione Tartaruga (stile LOGO)
<b>Finestra 'Codifica' ...</b>	Abilita la finestra per visualizzare la codifica automatica nei linguaggi previsti da Flowgorithm
<b>Copia immagine in memoria</b>	Copia l'immagine in memoria
<b>Esporta immagine in un file ...</b>	Esporta l'immagine area attiva in un file PNG
<b>Esporta immagine in formato vettoriale</b>	Esporta l'immagine area attiva in un file in formato vettoriale <b>.SVG</b> (Scalable Vector Graphics) o <b>.EMF</b> (Enhanced Metafile Format)

### *Esporta immagine in un file*

Selezionando la voce di menù **Esporta immagine in un file** viene aperta la seguente finestra. Se lasciamo tutto così, come ci viene proposto, e clicchiamo sul pulsante **Crea**, ci verrà proposto di dare un nome al file (con estensione PNG) e di selezionare in quale cartella salvarlo.

All'interno della finestra ci vengono proposte tre schede per la selezione di ulteriori impostazioni prima della creazione.

La prima scheda **Stile** ci ricorda quella che viene attivata dalla sequenza del menù **Modifica → Scegli** lo stile del diagramma vista precedentemente.



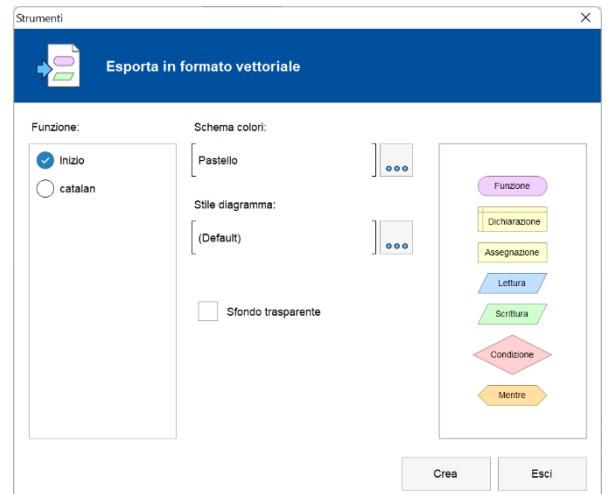
Il campo **Dimensioni** permette di impostare la grandezza del carattere. Un valore di 18 è considerato un valore adatto nella maggior parte dei casi.

Il campo **Margini** consente di impostare lo spazio in pixels intorno all'immagine (padding). Per Word è consigliato impostare tale spazio a 0.

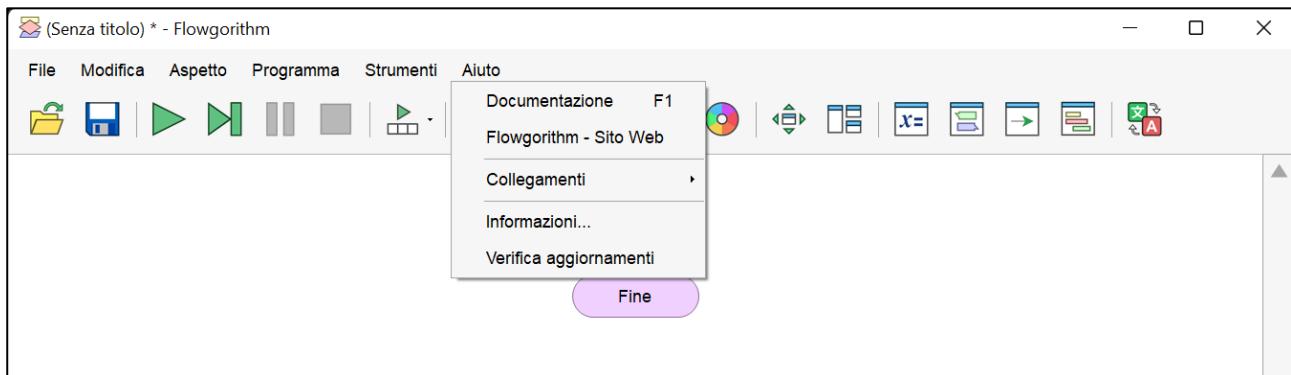
### *Esporta immagine in formato vettoriale*

Selezionando la voce di menù **Esporta immagine in formato vettoriale** viene aperta la seguente finestra.

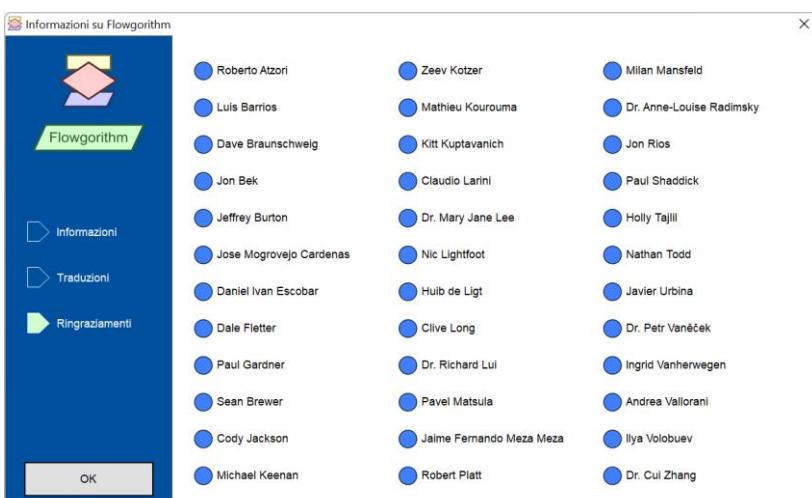
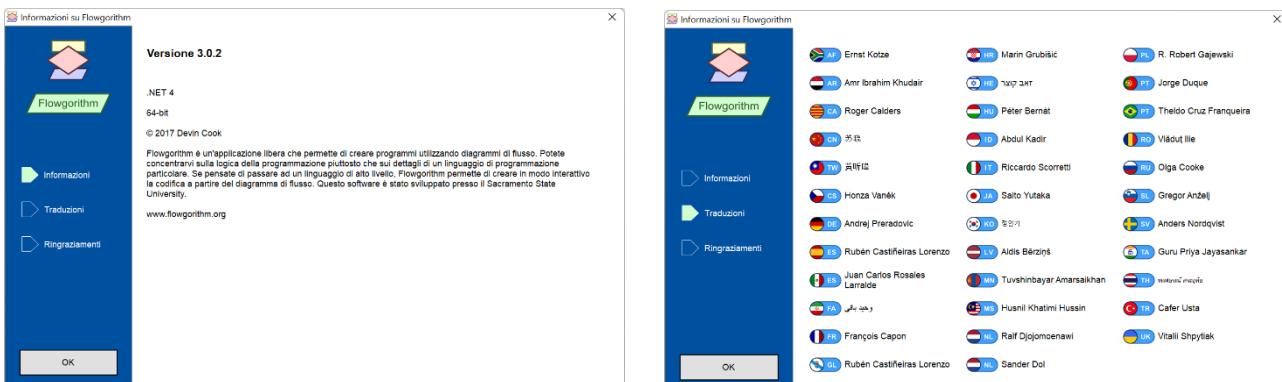
È possibile salvare l'immagine della funzione scelta in un file in formato .SVG (Scalable Vector Graphics) o in formato .EMF (Enhanced Meta File). Inoltre è possibile salvare le immagini anche con sfondo trasparente.



## Voce di menù "Aiuto"



Voce menù	Descrizione
<b>Documentazione</b>	Apre la pagina online relativa alla documentazione <a href="http://www.flowgorithm.org/documentation/">http://www.flowgorithm.org/documentation/</a>
<b>Flowgorithm – Sito Web</b>	Apre la pagina iniziale online relativa al sito Flowgorithm <a href="http://www.flowgorithm.org">http://www.flowgorithm.org</a>
<b>Collegamenti</b>	Apre un sottomenu di collegamenti all'Università di appartenenza dell'autore di Flowgorithm
<b>Informazioni</b>	Apre finestra informazioni sul software, traduzioni e ringraziamenti a chi ha partecipato all'evoluzione di questo software
<b>Verifica aggiornamenti</b>	Permette di controllare se esistono nuove versioni del software



## Le finestre di Flowgorithm

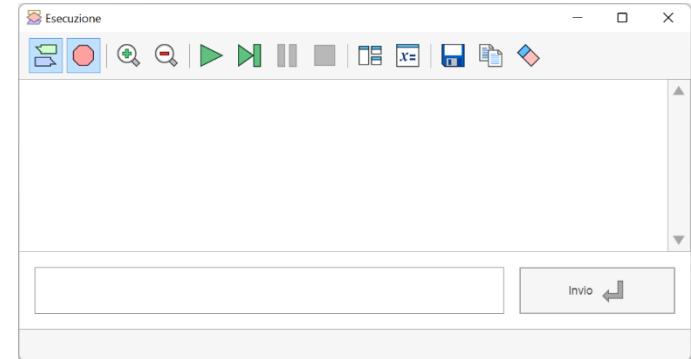
### Finestra "Esecuzione"

Selezionando la voce di menù **Finestra "Esecuzione"**



viene attivata una finestra utilizzata per monitorare l'esecuzione dell'algoritmo e per inserire da tastiera i valori necessari.

Il metodo classico per interagire con il computer è utilizzare la modalità "Console". Usando questo approccio, il programma visualizza in uscita su uno schermo solo testo e l'utente immette i dati utilizzando la tastiera. A volte la console permette al testo di cambiare i colori, ma, per la maggior parte, è testo bianco su sfondo nero.



Mentre la Console funziona, la sua interfaccia semplice può rendere difficile distinguere l'input dell'utente e l'output del programma. Così, invece di usare la schermata di solo testo, Flowgorithm trasforma il tutto in una tipica finestra di messaggio. Dal punto di vista dei programmatori, sono messaggi diretti e testuali.

I messaggi sono colorati in abbinamento alle forme di ingresso e di uscita utilizzati nel diagramma di flusso. Nella schermata a destra, l'input dell'utente è visualizzato in blu, mentre l'output del programma viene visualizzato in verde. Nella sua **barra degli strumenti** si possono notare delle icone che non compaiono nella barra degli strumenti della finestra principale.

Pulsante	Descrizione
A blue square with a white arrow pointing right and a purple square with a white arrow pointing left.	Permette la visualizzazione dei valori sotto forma di riquadri di dialogo (visione grafica)
A red octagon.	Permette la visualizzazione delle interruzioni
A green magnifying glass with a plus sign.	Aumenta ingrandimento
A green magnifying glass with a minus sign.	Diminisci ingrandimento
A green triangle pointing right.	Esegui interprete flowchart in automatico secondo la velocità impostata
A green triangle pointing right with a small vertical bar next to it.	Passo successivo modalità utente
Two vertical red bars.	Pausa nella modalità automatica
A solid red square.	Interrompi interprete flowchart
A window icon with a double-headed arrow.	Disposizione e selezione finestre
A window icon with an 'X' and an equals sign.	Finestra 'Variabili'
A blue floppy disk icon.	Salva il flowchart
A document icon with a blue border.	Copiare in memoria il contenuto della finestra
A diamond-shaped icon with a red border.	Cancellare il contenuto della finestra

### Finestra "Codifica"

Selezionando la voce di menù Finestra “Codifica”



si aprirà la finestra tramite la quale sarà possibile vedere la trasformazione del flowchart creato, codificato **in automatico** in uno dei tanti linguaggi di programmazione predisposti da Flowgorithm.

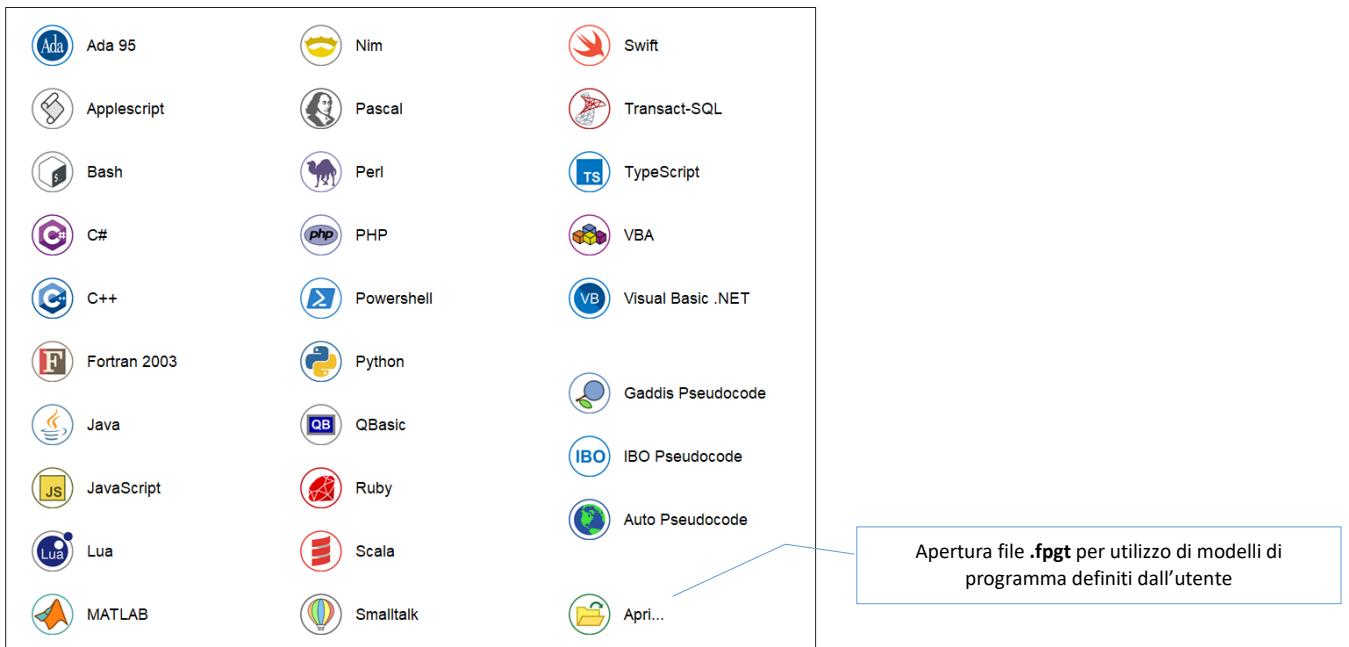
```

0 Funzione Inizio()
1   Dichiaro i: Intero
2
3   Per i = 0 a 9
4     Scrivi catalan(i)
5
6   Fine
7
8 Funzione catalan (n: Intero)
9   Dichiaro res: Intero
10  Dichiaro i: Intero
11
12  Se n <= 1
13    Assegna res = 1
14  Altrimenti
15    Assegna res = 0
16    Per i = 0 a n - 1
17      Assegna res = res + catalan(i) * catalan(n - i - 1)
18
19   Fine
20 Restituisce res: Intero

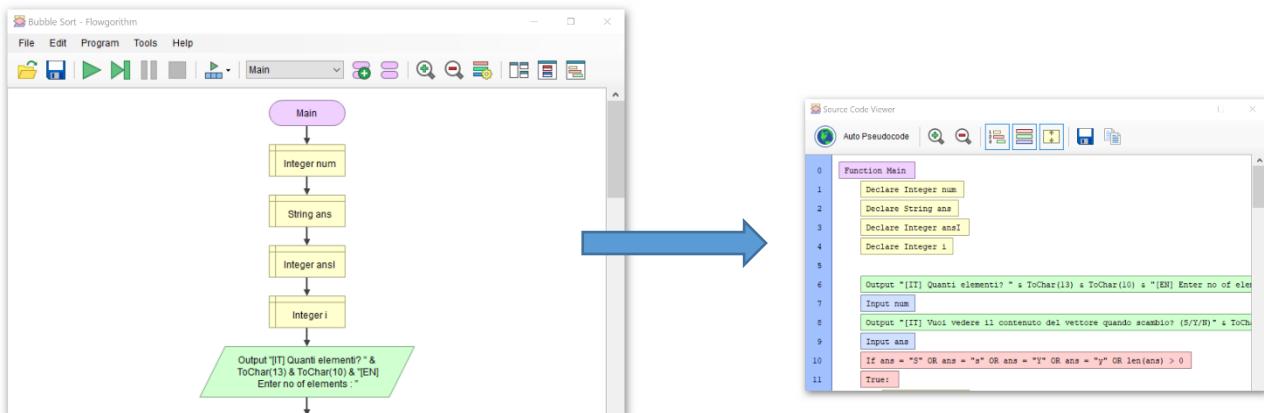
```

Pulsante	Descrizione
	Auto Pseudocode ▾ Permette la selezione della codifica in uno dei tanti linguaggi di programmazione
	Permette la visualizzazione dei numeri di linea
	Permette la visualizzazione delle istruzioni con colori
	Distanziamento linee del codice (interlinea)
	Aumenta ingrandimento
	Diminuisci ingrandimento
	Esegui interprete flowchart in automatico secondo la velocità impostata
	Passo successivo modalità utente
	Pausa nella modalità automatica
	Interrompi interprete flowchart
	Salva il flowchart
	Copiare in memoria il contenuto della finestra

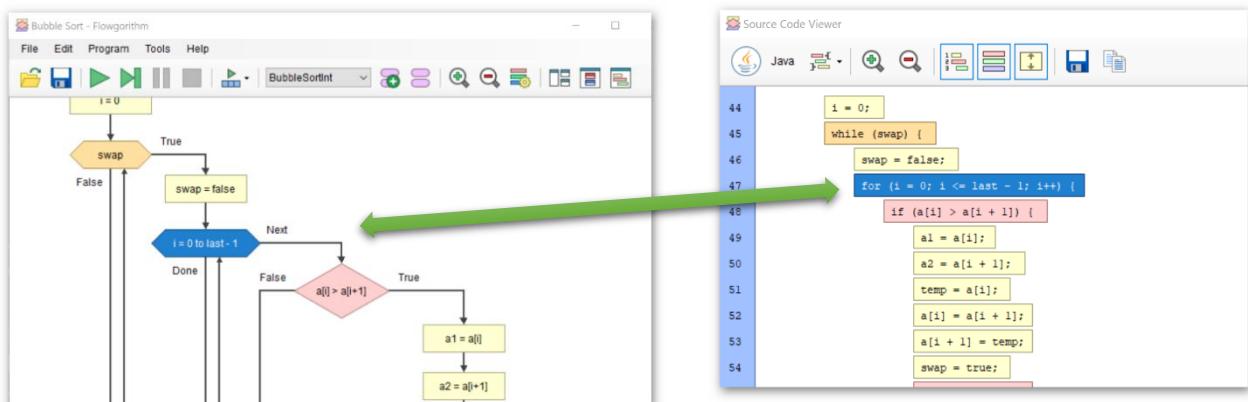
Il flowchart può essere codificato nei seguenti linguaggi di programmazione:



Il pulsante permette di disabilitare o abilitare l'evidenziazione delle istruzioni



Da sottolineare il fatto che questa funzionalità permette di individuare in maniera interattiva l'elemento del diagramma direttamente nella finestra codice. Infatti selezionando, ad esempio, una struttura del diagramma di flusso, verrà evidenziata la stessa all'interno della finestra codice di riferimento.



Il pulsante permette di rimuovere o aggiungere lo spazio tra le righe per una migliore lettura

```
i = 0;
while (swap) {
    swap = false;
    for (i = 0; i <= last - 1; i++) {
        if (a[i] > a[i + 1]) {
            a1 = a[i];
            a2 = a[i + 1];
            temp = a[i];
            a[i] = a[i + 1];
            a[i + 1] = temp;
            swap = true;
        }
    }
}
```

(attivato)

```
i = 0;
while (swap) {
    swap = false;
    for (i = 0; i <= last - 1; i++) {
        if (a[i] > a[i + 1]) {
            a1 = a[i];
            a2 = a[i + 1];
            temp = a[i];
            a[i] = a[i + 1];
            a[i + 1] = temp;
            swap = true;
        }
        if (ans == 1) {
    }
}
```

(disattivato)

Il pulsante permette di disabilitare o abilitare la numerazione delle righe

```
1 i = 0;
2 while (swap) {
3     swap = false;
4     for (i = 0; i <= last - 1; i++) {
5         if (a[i] > a[i + 1]) {
6             a1 = a[i];
7             a2 = a[i + 1];
8             temp = a[i];
9             a[i] = a[i + 1];
10            a[i + 1] = temp;
11            swap = true;
12        }
13    }
14}
```

(attivato)

```
i = 0;
while (swap) {
    swap = false;
    for (i = 0; i <= last - 1; i++) {
        if (a[i] > a[i + 1]) {
            a1 = a[i];
            a2 = a[i + 1];
            temp = a[i];
            a[i] = a[i + 1];
            a[i + 1] = temp;
            swap = true;
        }
        if (ans == 1) {
            System.out.println("SWAP i: " + a1);
            showArrayIn(a);
        }
    }
}
```

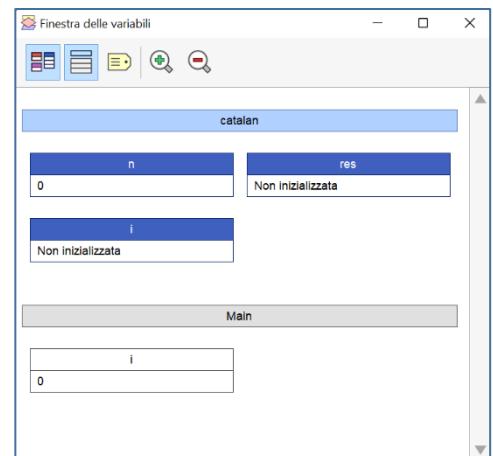
(disattivato)

Se avete intenzione di imparare un nuovo linguaggio di alto livello, questa caratteristica è di grande aiuto. Il codice viene evidenziato utilizzando lo stesso colore utilizzato per le forme corrispondenti utilizzate nel diagramma di flusso. Questo permette, visivamente, di vedere la correlazione tra il codice e il diagramma di flusso. Inoltre, se si evidenzia una forma nel diagramma di flusso, o fate un passo con il vostro programma, il codice sorgente verrà evidenziato nello stesso punto.

### Finestra "Variabili"

Selezionando la voce di menù **Finestra "Variabili"**  si aprirà la finestra relativa. La finestra di controllo delle variabili viene utilizzata per tenere traccia di come le variabili cambiano mano che il programma viene eseguito. Quindi, se si è arrivati ad un punto del programma, la finestra mostrerà ogni variabile e il suo valore corrente. Questo non è limitato solo a variabili fondamentali. Anche gli array verranno visualizzati.

Ogni variabile ha un colore codificato in base al suo tipo di dati. A prima vista, si può dire esattamente che tipo di dati vengono memorizzati e focalizzarsi su un determinato tipo di dato.



year	1947
ratio	1.618
name	Sacramento State
done	True

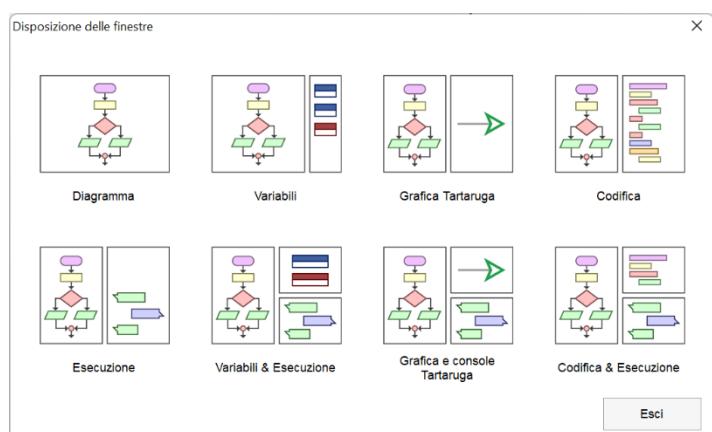
- Gli Interi sono visualizzati in blu.
- I numeri Reali sono visualizzati in viola.
- Le variabili di tipo Stringa sono visualizzate in rosso.
- Le variabili Booleane sono visualizzate in verde scuro.

Pulsante	Descrizione
	Visualizzazione in colonne
	Visualizzazione pila (programma principale + funzioni)
	Visualizzazione etichette
	Aumenta ingrandimento
	Diminuisci ingrandimento

### Disposizione finestre

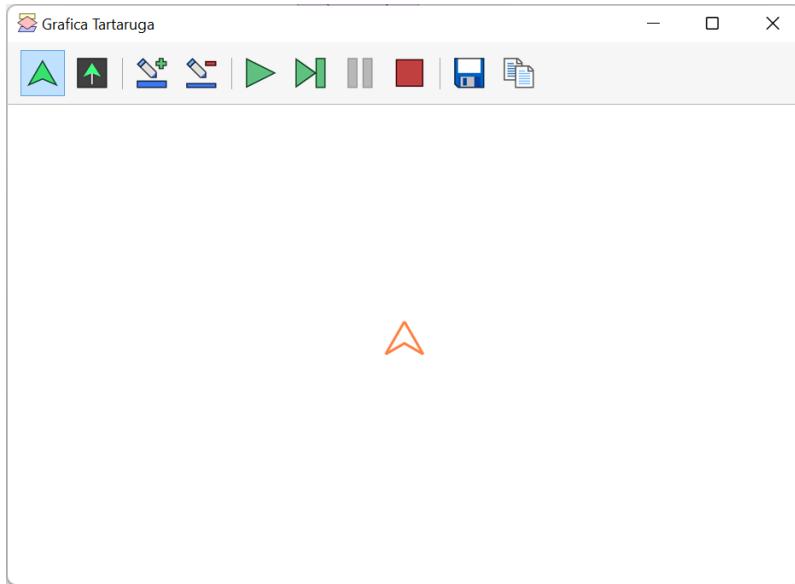
Selezionando la voce di menù **Disposizione**

 si aprirà la finestra per dare la possibilità di una impostazione delle finestre definita dall'utente secondo le proprie necessità.



### *Finestra grafica "Tartaruga"*

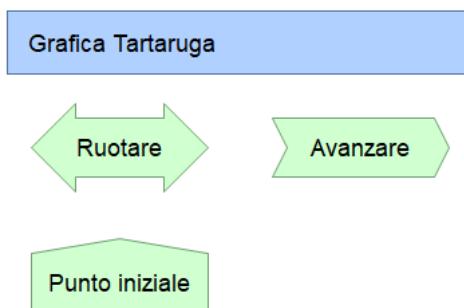
Selezionando la voce di menù **Disposizione finestre**  si aprirà la finestra per dare la possibilità dell'uso della grafica Tartaruga

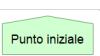


La versione 3 introduce la grafica della Tartaruga in Flowgorithm. L'idea deriva dal classico linguaggio di programmazione LOGO, sviluppato nel 1967. Questa modalità permette di creare immagini con le semplici azioni della "Tartaruga" - un "pennello" grafico che può essere ruotato e spostato per creare immagini. L'obiettivo è insegnare i concetti di programmazione attraverso i passaggi sequenziali necessari per creare un'immagine.

L'introduzione della grafica Tartaruga è uno strumento aggiuntivo per insegnare la programmazione. Ad esempio, si potrebbe chiedere allo studente di disegnare un rettangolo, prima attraverso una sequenza di passaggi, poi, usando l'iterazione.

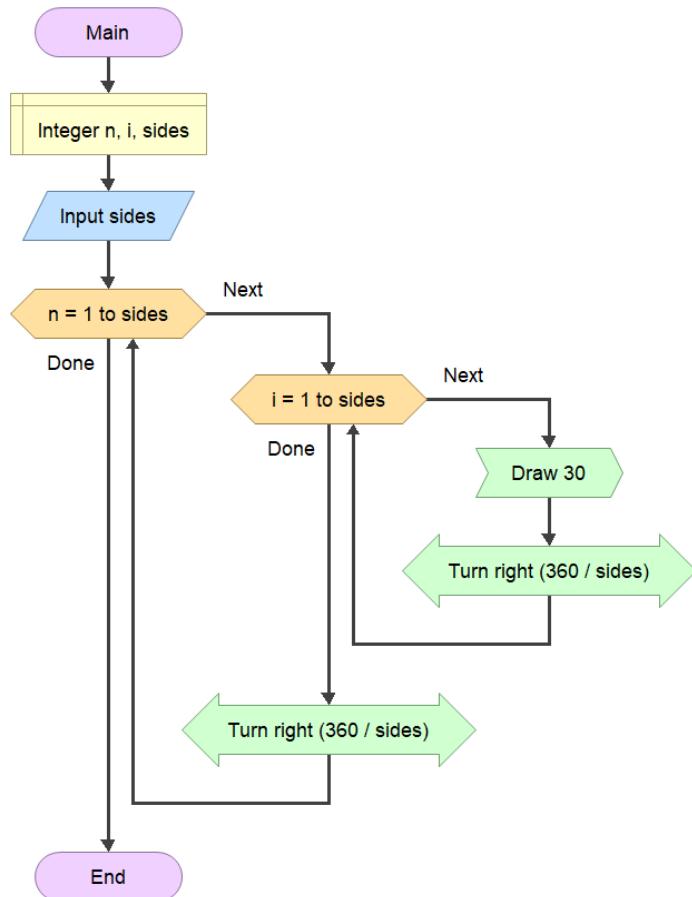
In Flowgorithm sono state altre tre forme:



 Ruotare	Cambia l'angolo ruotando la Tartaruga a sinistra o a destra di un numero specificato di gradi. Questo ricorda i comandi Logo LT (svolta a sinistra) e RT (svolta a destra).
 Avanzare	Sposta la tartaruga della distanza specificata. Questo ricorda i comandi Logo PU (penna su), PD (penna giù), FD (avanti) e BK (indietro).
 Punto iniziale	Piporta la tartaruga alla posizione iniziale e imposta l'angolo a 90 gradi. Questo è lo stesso del comando Logo HOME

Alcune note sulla finestra grafica della tartaruga:

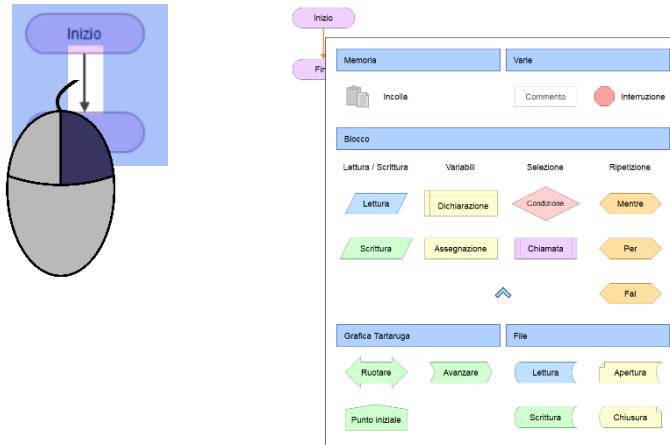
- I comandi grafici vengono memorizzati, quindi l'immagine può essere ridimensionata e vettoriale se la finestra viene ridimensionata. Ciò significa anche che il disegno può essere esportato in SVG o in un programma Logo.
- Il buffer, utilizzato per memorizzare l'immagine, è limitato e cancellerà gli elementi se diventa eccessivamente grande (cosa che potrebbe fare uno studente). In realtà, gli studenti non dovrebbero raggiungere questo limite.
- Il campo di disegno ha una "estensione" (da -1.000.000 e 1.000.000) sia per l'asse x che per y.



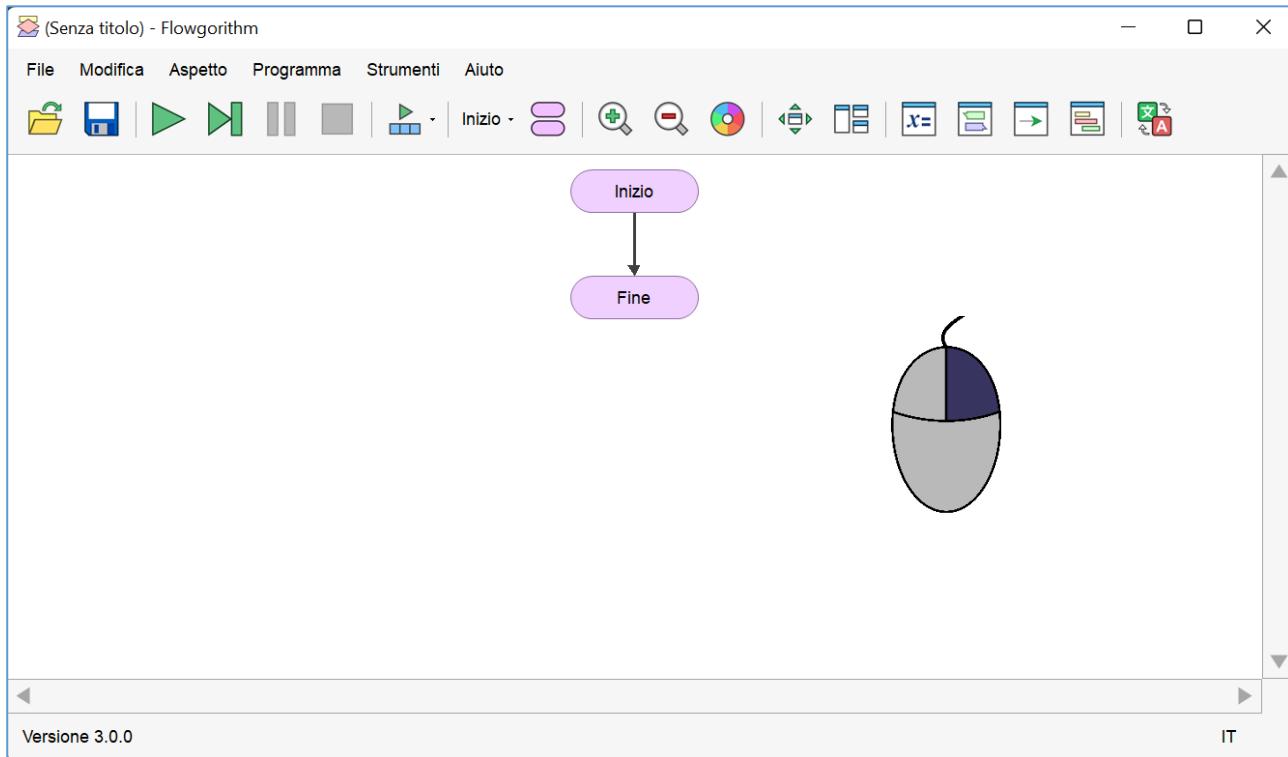
## Menù contestuale

Il menù contestuale può essere attivato utilizzando il tasto destro del mouse su specifiche posizioni.

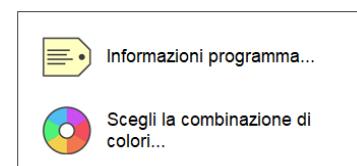
- Se si clicca, col tasto destro del mouse, sulla freccia di flusso apparirà un menu pop-up mostrando tutte le figure che possono essere inserite



- Se si clicca, col tasto destro del mouse, sulla parte vuota della finestra di edizione



si attiverà il menù pop-up a destra attraverso il quale è possibile passare a menù già trattati precedentemente.



## Specifiche

- Array mono dimensionali
- Cicli con controllo in testa e in coda
- Cicli enumerativi (FOR)
- Dichiarazione esplicita delle variabili. I diagrammi di flusso generalmente usano dichiarazione implicita (essendo abbastanza minimalisti). Tuttavia, praticamente tutti i linguaggi di programmazione utilizzano dichiarazioni esplicite. Quindi, è una buona idea per gli studenti impararlo fin dall'inizio.
- Ricorsione controllata. Flowgorithm è fornito di uno stack interno (piuttosto che lo stack del sistema). Se lo studente crea accidentalmente una chiamata ricorsiva infinita, il programma non si bloccherà. Si riceverà, al contrario, un messaggio di errore.
- Supporta operatori sia della famiglia dei linguaggi di programmazione come C che della famiglia del BASIC. Si possono utilizzare qualunque di questi operatori indipendentemente dal linguaggio di programmazione che si vuole imparare.
- Sono supportate funzioni e procedure create dall'utente che possono o meno restituire un valore.
- Oltre 20 funzioni predefinite.

## E ancora...

- Sono supportati diversi stili di rappresentazione come il tipo classico, IBM, SDL, e altri...
- I diagrammi di flusso possono essere stampati sia a colori che in bianco e nero.
- I diagrammi di flusso possono essere salvati in un file PNG sia a colori che in bianco e nero.
- L'editor possiede una caratteristica chiamata "Layout Windows". È progettato per aiutare a visualizzare il contenuto senza alcun intervento manuale sulla finestra.
- I programmi sono salvati in XML.

# Il mio primo diagramma di flusso

Questo breve tutorial, serve creare un semplice programma chiamato "Ciao Mondo". Si tratta di un programma tradizionale per neofiti che visualizza "Ciao, mondo!" sullo schermo.

Quando si avvia un nuovo diagramma di flusso, vedrete due rettangoli arrotondati chiamati simboli "terminali". Questi simboli rappresentano l'inizio e la fine del programma.

Molti diagrammi iniziano il flusso visualizzando il testo "Start". Flowgorithm, invece, usa il testo "Main". La maggior parte dei linguaggi di programmazione iniziano con "Main" e Flowgorithm ... altrettanto.

Tutto, in un diagramma di flusso, è rappresentato da una figura geometrica. Potrai aggiungere le figure tra la figura principale e quella finale.

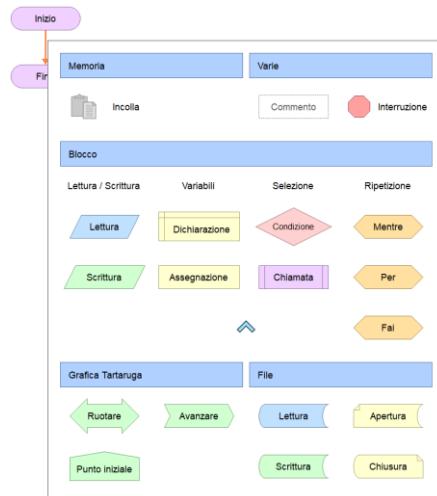
Per aggiungere una figura, spostare il puntatore del mouse su una linea. Se è possibile aggiungere una figura, la linea diventa di colore arancione.

Fare doppio clic o clicca col pulsante destro del mouse per aggiungere una figura.

Un menu pop-up apparirà mostrando tutte le figure che possono essere inserite.

Nei diagrammi di flusso, ogni azione che il computer può eseguire è rappresentata da una figura diversa. Ad esempio, le figure di ingresso e di uscita sono rappresentate con parallelogrammi.

Si può utilizzare anche il "copia e incolla". Nell'immagine qui sotto è disegnata in grigio perché non si è ancora copiato niente in memoria.

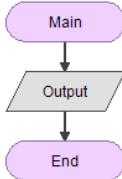


Il programma "Ciao Mondo" necessita di una figura per l'output. Ciò permetterà di visualizzare l'informazione nello schermo del computer.

Spostare il mouse sopra il parallelogramma verde con la scritta interna "Output": ora verrà evidenziata con un rettangolo blu.

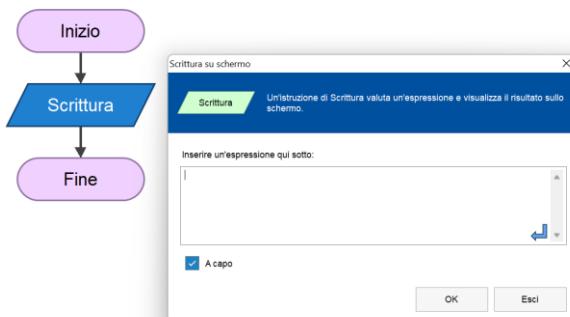
Cliccare sulla figura col pulsante sinistro del mouse e la figura verrà inserita nel diagramma.

Appena inserita, la figura apparirà colorata di grigio. In Flowgorithm, il colore grigio indica che la figura è "incompleta". Ciò significa che la figura ha necessità di altra informazione per poter funzionare.



È necessario indicare a Flowgorithm cosa volete che visualizzi. Ciò può essere qualsiasi cosa: da un risultato di un calcolo ad un messaggio.

Eseguendo un doppio click sulla figura si entra nella modalità modifica.



Apparirà una finestra intitolata "Scrittura su schermo". Con questa finestra sarà possibile indicare cosa far apparire nello schermo del computer.

Nella finestra, scrivere "Ciao mondo!". Attenzione: è necessario aggiungere anche gli apici. Nei linguaggi di programmazione ciò viene chiamata String (stringa).

Cliccare sul pulsante OK per continuare. Se viene commesso un errore, Flowgorithm ci avvisa subito. Doppio click sulla figura per correggere l'errore.



La colorazione in verde della figura di output indicherà la corretta creazione.

Poiché tutte le figure sono colorate nella maniera corretta, ora è possibile eseguire quanto creato.

Sulla barra principale clicca sul pulsante  per eseguire il programma.

Apparirà una nuova finestra chiamate "Console Screen" usata per visualizzare le informazioni e per inserire i dati.

Hai realizzato il tuo primo programma. Congratulazioni!

## Tipi di dato

### Integer (Intero)

Il tipo di dati Integer è uno dei tipi più comunemente utilizzati nella programmazione. Una variabile di tipo integer può memorizzare un numero intero positivo o negativo, ma non può memorizzare valori frazionari. Così, può memorizzare valori quali 5, 42, 1947, ma non può memorizzare numeri come 3.2, 4.5, ecc ...

Integer
1947

La finestra di controllo variabili visualizza i numeri interi in blu.

### Real (Reale)

Il tipo di dati Real può memorizzare un numero qualsiasi, sia i numeri interi che quelli frazionari. In molte lingue, questo è chiamato un Double dopo lo standard implementazione noto come "virgola mobile a doppia precisione".

Real
1.618

La finestra di controllo variabili visualizza i reali in viola.

### String (Stringa)

Il tipo di dati String viene utilizzata per memorizzare tutti i dati testuali. Questo include parole, lettere, o qualsiasi altra cosa si potrebbe inviare con un messaggio di testo. Nella programmazione, il testo viene delimitato da doppi apici. Ad esempio: "CSU, Sacramento", "computer", e "Anno 1947" sono tutte delle stringhe.

String
Sacramento State

La finestra di controllo variabile visualizza le stringhe in rosso.

### Boolean (Booleano o Logico)

Il tipo di dati booleano è in grado di memorizzare sia "vero" o "falso". Questi sono la base del processo decisionale in un programma per computer.

Boolean
True

La finestra di controllo variabile mostra booleani in grigio verde.

# Identificatori

Ogni volta che si definisce una funzione o una variabile, viene dato un nome univoco chiamato "identificatore". Per evitare che gli identificatori possano essere confusi con altri elementi in una espressione, esse devono seguire una convenzione. Ogni linguaggio di programmazione ne possiede una e possono differire da un linguaggio all'altro.

In Flowgorithm, gli identificatori devono seguire le seguenti regole:

- Devono iniziare con una lettera.
- Dopo la prima lettera, l'identificatore può contenere lettere o numeri.
- Non sono permessi gli spazi.
- Non possono essere parole riservate o parole già definite in Flowgorithm.

Altre note:

- I linguaggi di programmazione come il Visual Basic e il C permettono anche l'uso del carattere "\_" (underscore). Flowgorithm, al contrario, non lo accetta.
- Gli identificatori non sono case-sensitive, per cui "CASA" è uguale a "casa".

# Parole chiave

Molte parole usate nei linguaggi di programmazione si sovrappongono alla convenzione di denominazione utilizzata per gli identificatori. In questi casi, la parola è "riservata" e non può essere utilizzata per gli identificatori. Inoltre, in molti linguaggi di programmazione funzioni e altre costanti sono predefinite. Anche queste non possono essere utilizzate.

# Operatori

Le espressioni utilizzano operatori usati nei due principali linguaggi di programmazione: la famiglia dei linguaggi di programmazione afferenti al "BASIC" e la famiglia dei linguaggi di programmazione afferenti al "C" (che include C, Java, C#,...). Flowgorithm supporta i simboli utilizzati in matematica (usando i valori Unicode)

Poiché sono supportate entrambe le famiglie, siamo in presenza di un numero ridondante di operatori. Questi sono:

Operator	C Family	BASIC Family	Mathematics (Unicode)
Equality	==	=	=
Inequality	!=	<>	≠
Less Than or Equal	<=	<=	≤
Greater Than Or Equal	>=	>=	≥
Logical Not	!	not	¬
Logical And	&&	and	∧
Logical Or		or	∨
Multiply	*	*	×
Divide	/	/	÷
Modulo	%	mod	(no symbol)

Flowgorithm fornisce anche la possibilità d'uso di alcuni operatori tipici del Visual Basic poiché sono utili, semanticamente ben definiti:

Operatore Visual Basic	Descrizione
&	Concatenazione tra stringhe
^	potenza

In Java e in C#, l'operatore "+" è utilizzato sia per la concatenazione di stringhe sia come operatore nelle addizioni. Ciò potrebbe trarre in inganno. In Flowgorithm, l'operatore "+" è utilizzato come operatore aritmetico. Per la concatenazione viene usato il simbolo "&" (ampersand) e per l'operazione di esponenziazione, il simbolo "^"

## Precedenze

La tabella seguente evidenzia il livello di precedenza degli operatori aritmetici e logici, dal più alto (valutato per prima) al più basso (valutato per ultimo).

Livello	Nome	Operazione	Note
8	Operatore unario	- ! not $\neg$	In Visual Basic 'not' precede tutti gli operatori relazionali
7	Esponente (potenza)	$^$	Il simbolo non è utilizzato in C# o Java
6	Operatori aritmetici	$* \times / \div$ $\% \text{ mod}$	La divisione ritorna sempre un numero reale
5	Operatori aritmetici	$+ -$	Il simbolo '+' è utilizzato solo con i numeri
4	Concatenazione	&	C# e Java usano in maniera ambigua l'operatore '+'
3	Operatori relazionali	$> < \geq \leq \leq \leq$ $\text{==} \neq \text{!=} \text{<>} \neq$	
2	E (logico)	$\&\& \text{ and } \wedge$	
1	O (logico)	$\  \text{ or } \vee$	

## Esempi

Espressione	Soluzione	Note
$1 + 3 ^ 2$	10	
$10 * 2 + 5 * 6$	50	E' come scrivere $(10 * 2) + (5 * 6)$
$7 * (4 - 1)$	21	Le parentesi modificano le precedenze
$6 / 3 + 2$	4	E' come scrivere $(6 / 3) + 2$
$10 \text{ MOD } 3$	1	MOD restituisce il resto della divisione tra interi
$10 \% 3$	1	Come la precedente ma per la famiglia C

# Funzioni predefinite

## Matematiche

Funzione	Descrizione	Dalla versione..
<b>Abs(n)</b>	Valore assoluto	
<b>Arcsin(n)</b>	Arcoseno di n (radiani)	1.7
<b>Arccos(n)</b>	Arcocoseno di n (radiani)	1.7
<b>Arctan(n)</b>	Arcotangente di n (radiani)	
<b>Cos(n)</b>	Coseno di n (radiani)	
<b>Sin(n)</b>	Seno di n (radiani)	
<b>Tan(n)</b>	Tangente di n (radiani)	
<b>Int(n)</b>	Parte intera di un numero reale	
<b>Ln(n)</b>	Logaritmo naturale	
<b>Log(n)</b>	Logaritmo naturale, come Ln(n)	
<b>Log10(n)</b>	Logaritmo in base 10	
<b>Sgn(n)</b>	Segno matematico (-1 se n è negativo, 0 se zero, 1 se positivo)	
<b>Sqrt(n)</b>	Radice quadrata di n	

## Stringa

Funzione	Descrizione
<b>Len(s)</b>	Restituisce la lunghezza (intero) di una stringa es: Len("casa") → 4
<b>Char(s, i)</b>	Restituisce il carattere alla posizione i della stringa s. La posizione parte da 0. es: Char("casa", 1) → "a"

## Conversione tipo di dati

Funzione	Descrizione	Dalla versione..
<b>ToChar(n)</b>	Converte un codice carattere n in un carattere es: ToChar(65) → "A"	1.5
<b>ToCode(c)</b>	Converte un carattere c in un codice carattere (intero) es: ToCode("A") → 65	1.8
<b>ToFixed(r, i)</b>	Converte un numero reale r in una stringa con i cifre decimali es: ToFixed(123.456, 1) → "123.4"	1.8
<b>ToInteger(s)</b>	Converte una stringa con caratteri numerici in un intero es: Tolnteger("123") → 123	
<b>ToReal(s)</b>	Converte una stringa con caratteri numerici in un reale es: ToReal("123.5") → 123.5	
<b>ToString(n)</b>	Converte un numero in stringa es: ToString(123.5) → "123.5"	

## Altre

Funzione	Descrizione	Dalla versione..
<b>EOF()</b>	Restituisce TRUE se si raggiunge la fine del file. Questa funzione è usata per la lettura di un file aperto	
<b>Random(n)</b>	Restituisce un numero intero tra <b>0 e n - 1</b>	
<b>Size(a)</b>	Restituisce la grandezza (numero degli elementi) di un vettore (array)	1.7

## Costanti predefinite

La semantica di Flowgorithm prevede tre costanti predefinite. “True” e “False” spesso utilizzate per inizializzare le variabili booleane e “Pi”, costante normalmente utilizzata in matematica.

Tabella Operatori

Costanti	Note
<b>True</b>	Valore booleano (logico) per Vero
<b>False</b>	Valore booleano (logico) per Falso
<b>Pi</b>	Costante matematica (P greco $\pi = 3.1415$ )
<b><math>\pi</math></b>	Costante matematica (P greco $\pi = 3.1415$ ) come sopra

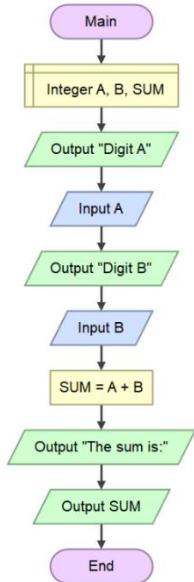
# Formato File Sorgente

Molte applicazioni salvano i dati in un file binario (di difficile lettura). Ciò ha senso se i dati sono molto complessi e se si vuole minimizzare la grandezza del file attraverso l'uso della compressione.

I programmi scritti con Flowgorithm sono molto semplici. Non esistono dati di grande complessità da dover essere compressi. I dati vengono salvati in un file di testo proprio come se fossero scritti in Java, C#, ecc... In questo modo, piuttosto che avere file binari complessi, questi ultimi sono di facile lettura e portabili verso altri sistemi. L'idea è che usando questa applicazione questi file possano essere letti da altri strumenti (tipo editor). Perciò i files vengono salvati in formato XML (Extended Markup Language).

## Esempio

Prendiamo come esempio il seguente flowchart corrispondente all'esercizio "Somma tra due numeri":



Quando l'utente salva il suo esercizio con un determinato nome, Flowgorithm genera un file. Il file generato da Flowgorithm (file sorgente) ha estensione .FPRG, ma non è un file proprietario cioè non è un file che può essere letto solo dal software di appartenenza.

Proviamo ad aprire il file .FPRG corrispondente all'esercizio di cui sopra utilizzando non Flowgorithm ma un editor di testo come ad esempio WordPad. Ecco cosa potremo vedere.

```
<?xml version="1.0"?>
<flowgorithm fileversion="2.11">
    <attributes>
        <attribute name="name" value="" />
        <attribute name="authors" value="Roberto" />
        <attribute name="about" value="" />
        <attribute name="saved" value="2019-02-07 08:45:07" />
        <attribute name="created" value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0wMi0wNzsiMDg6NDI6NDUgIjszMDU2" />
        <attribute name="edited" value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0wMi0wNzsiMDg6NDU6MDcgIjsxOzMxNjU" />
    </attributes>
    <function name="Main" type="None" variable="">
        <parameters/>
        <body>
            <declare name="A, B, SUM" type="Integer" array="False" size="" />
            <output expression=""Digit A"" newline="True" />
            <input variable="A" />
            <output expression=""Digit B"" newline="True" />
            <input variable="B" />
            <assign variable="SUM" expression="A + B" />
            <output expression=""The sum is:"" newline="True" />
            <output expression="SUM" newline="True" />
        </body>
    </function>
</flowgorithm>
```

Forse a molti questo codice risulterà incomprensibile ma, anche per costoro, potrebbe essere utile sapere alcune informazioni.

Per prima cosa si nota che l'autore, D. Cook, ha utilizzato il linguaggio XML (eXtended Markup Language) che è una estensione del linguaggio HTML (Hyper Text Markup Language), il linguaggio base che viene utilizzato per la generazione delle pagine web.

Perché l'autore ha utilizzato l'XML? Una prima risposta potrebbe essere perché è intelligibile cioè, nonostante sia un codice sconosciuto ai più, si riesce a interpretare anche se vagamente. E poi, con l'XML è possibile creare nuovi tag secondo dell'intendimento dell'autore.

Vediamolo un po' più nel dettaglio.

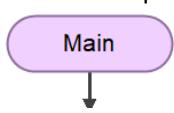
La parte iniziale:

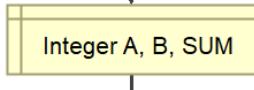
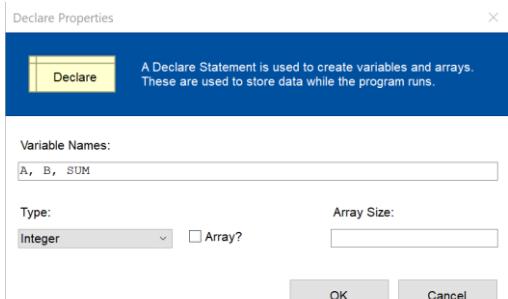
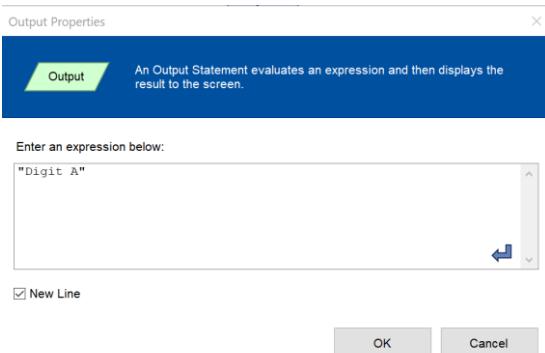
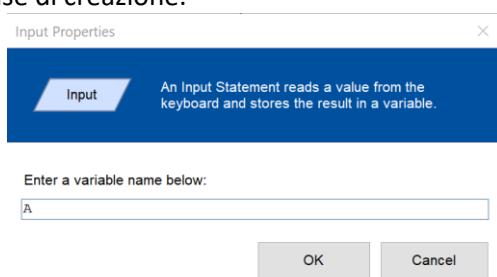
```
<?xml version="1.0"?>
<flowgorithm fileversion="2.11">
    <attributes>
        <attribute name="name" value="" />
        <attribute name="authors" value="Roberto" />
        <attribute name="about" value="" />
        <attribute name="saved" value="2019-02-07 08:45:07 " />
        <attribute name="created"
value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0wMi0wNzsiMDg6NDI
6NDUgIjszMDU2" />
        <attribute name="edited"
value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0wMi0wNzsiMDg6NDU
6MDcgIjsxOzMxNjU=" />
    </attributes>
```

Vediamo più nel dettaglio.

<?xml version="1.0"?>	Indica la versione del linguaggio XML utilizzata
<flowgorithm fileversion="2.11">	Indica la versione di Flowgorithm utilizzata per la creazione del file in oggetto
<attributes>         <attribute name="name" value="" />         <attribute name="authors" value="Roberto" />         <attribute name="about" value="" />         <attribute name="saved" value="2019-02- 07 08:45:07 " />         <attribute name="created" value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0w Mi0wNzsiMDg6NDI6NDUgIjszMDU2" />         <attribute name="edited" value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0w Mi0wNzsiMDg6NDU6MDcgIjsxOzMxNjU=" />     </attributes>	Questa parte è utilizzata per memorizzare alcune informazioni da associare al file sorgente (.FPRG) come Autore, Data di creazione, e qualche altra informazione utile a Flowgorithm.

A partire fin da ora, le linee di codice avranno un riferimento più “familiare”.

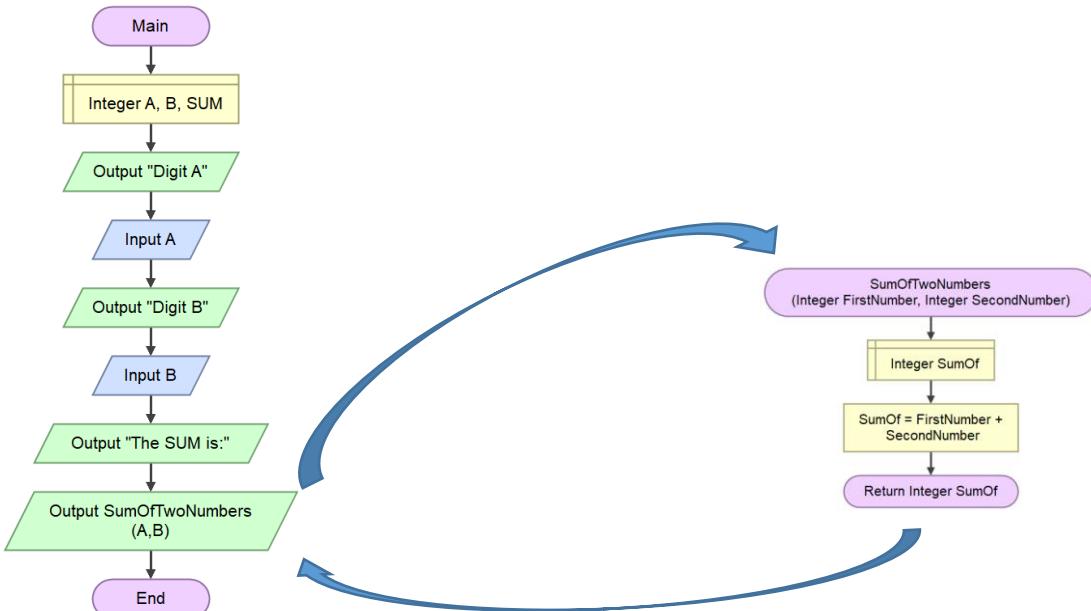
<function name="Main" type="None" variable="">	<p>È l'inizio di una funzione (function) particolare in quanto è la prima funzione e questa corrisponde al</p>  <p>Il tag "&lt;function ..&gt;" ha un suo tag di chiusura   &lt;/function&gt;</p>
--	--

<parameters/>	La funzione principale "Main" non ha parametri. Questo tag assume forma diversa nel caso di funzioni con parametri.
<body>	Il corpo della funzione e </body> il fine tag
<declare name="A, B, SUM" type="Integer" array="False" size="" />	<p>La dichiarazione delle variabili Nel flowchart:</p>  <p>In fase di creazione è facile individuare le singole voci di appartenenza all'interno del tag &lt;declare ...&gt;:</p> 
<output expression="\"Digit A\"" newline="True"/>	<p>Nel flowchart:</p>  <p>In fase di creazione:</p> 
<input variable="A"/>	<p>Nel flowchart:</p>  <p>In fase di creazione:</p> 
<output expression="\"Digit B\"" newline="True"/>	Nel flowchart:
<input variable="B"/>	Nel flowchart:

<assign variable="SUM" expression="A + B"/>	<p>Nel flowchart:</p>
<output expression="The sum is:" newline="True"/>	<p>In fase di creazione:</p>
<output expression="SUM" newline="True"/>	<p>Nel flowchart:</p>
</body>	Fine tag del tag <body>
</function>	Fine tag del tag <function>
</flowgorithm>	Fine tag del tag <flowgorithm> (termine esercizio)

Ovviamente possono essere presenti altri tag diversi da quelli visti fino ad ora e corrispondenti a particolari operazioni / funzioni di Flowgorithm.

Per capire bene qual è l'importanza (per l'utente generico) dell'uso dell'XML, proviamo a vedere il sorgente dell'esercizio "Somma tra due numeri" svolto utilizzando una funzione nuova creata dall'utente che svolge, appunto, la somma tra due numeri. L'esercizio si compone di due parti: una parte principale (MAIN) e una parte secondaria (FUNCTION) richiamata all'interno del MAIN.



Vediamo il file sorgente:

```
<?xml version="1.0"?>
<flowgorithm fileversion="2.11">
    <attributes>
        <attribute name="name" value="" />
        <attribute name="authors" value="Roberto" />
        <attribute name="about" value="" />
        <attribute name="saved" value="2019-02-07 04:04:45 " />
        <attribute name="created" value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0wMi0wNzsiMDg6NDI6NDUgIjszMDU2" />
        <attribute name="edited" value="Um9iZXJ0bztERVNLVE9QLVJPQkVSVE87MjAxOS0wMi0wNzsiMDQ6MDQ6NDUgIjszOzMxNjA=" />
    </attributes>
    <function name="Main" type="None" variable="">
        <parameters/>
        <body>
            <declare name="A, B, SUM" type="Integer" array="False" size="" />
            <output expression="Digit A" />
            <newline value="True" />
            <input variable="A" />
            <output expression="Digit B" />
            <newline value="True" />
            <input variable="B" />
            <output expression="The SUM is:" />
            <newline value="True" />
            <output expression="SumOfTwoNumbers (A, B)" />
            <newline value="True" />
        </body>
    </function>
    <function name="SumOfTwoNumbers" type="Integer" variable="SumOf">
        <parameters>
            <parameter name="FirstNumber" type="Integer" array="False" />
            <parameter name="SecondNumber" type="Integer" array="False" />
        </parameters>
        <body>
            <declare name="SumOf" type="Integer" array="False" size="" />
            <assign variable="SumOf" expression="FirstNumber + SecondNumber" />
        </body>
    </function>
</flowgorithm>
```

Oltre ad aver già spiegato prima parte della sua composizione, all'interno del file .FRPG (in XML) è presente una funzione in più rispetto a quella principale (MAIN): è la funzione "SumOfTwoNumbers" posizionata subito dopo la funzione MAIN.

```
<function name="SumOfTwoNumbers" type="Integer" variable="SumOf">
    <parameters>
        <parameter name="FirstNumber" type="Integer" array="False" />
        <parameter name="SecondNumber" type="Integer" array="False" />
    </parameters>
    <body>
        <declare name="SumOf" type="Integer" array="False" size="" />
        <assign variable="SumOf" expression="FirstNumber + SecondNumber" />
    </body>
</function>
```

Si nota, ora, la presenza di parametri (non presenti nella funzione MAIN).

Questa funzione, così costruita, puo' essere copiata dal file XML e incollata all'interno di un altro file XML (altro esercizio) che puo' far uso di questa funzione subito prima del tag <flowgorithm>. Dopo aver salvato il tutto e riaperto il file col software Flowgorithm, la funzione verrà posizionata nella zona funzioni perfettamente funzionante e pronta all'uso all'interno del nostro flowchart.

**In questa maniera è possibile costruire una libreria (un file di tipo testuale .TXT) di funzioni da copiare e incollare in quegli esercizi dove fosse necessario il loro utilizzo.**

# Istruzioni

## Assegnazione

Figura associata

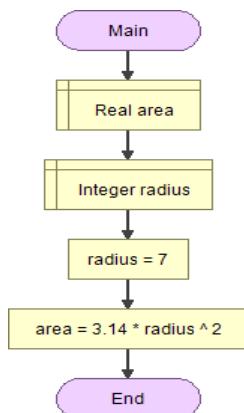
Assignment

Cosa fa

Un comando di assegnazione, calcola una espressione e registra il risultato in una variabile.

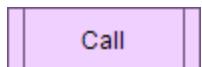
Esempio

Nel seguente esempio vengono dichiarate due variabili: "area" (che conterrà numeri reali) e "radius" (che conterrà numeri interi). Viene poi utilizzato il comando di assegnazione per impostare a 7 la variabile 'radius'. Per finire, nella variabile "area" verrà assegnato il calcolo dell'area di un cerchio.



## Chiamata di Funzione o Procedura

Figura associata



Cosa fa

Un comando di chiamata trasferisce il controllo ad una funzione o ad una procedura. Per veicolare le informazioni in una funzione o in una procedura, si usano i parametri associati alla funzione o alla procedura e ciò può essere portato a termine utilizzando o la modalità del passaggio per valore o la modalità del passaggio per riferimento.

## Commento

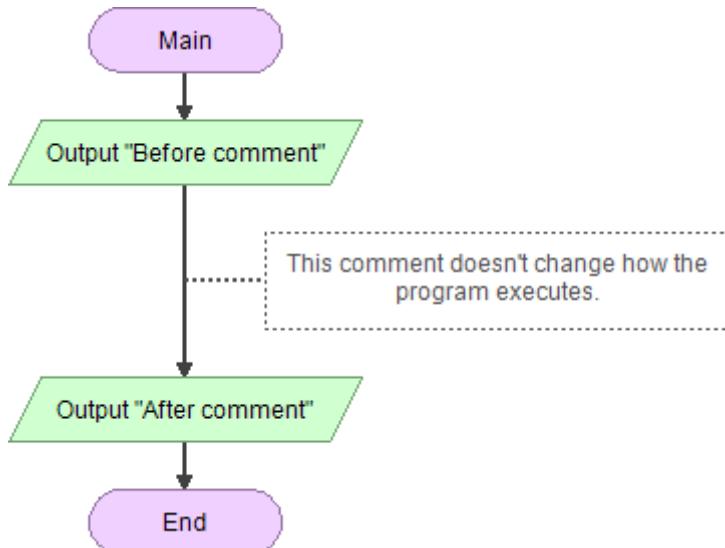
Figura associata



Cosa fa

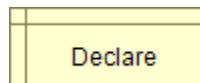
I commenti non interferiscono con l'esecuzione del programma. Sono utilizzati per documentare meglio le azioni intraprese e/o gli intendimenti dell'autore. Questi possono includere: la logica di un ciclo, regole, cambiamenti, ecc...

Esempio



## Dichiarazione

Figura associata

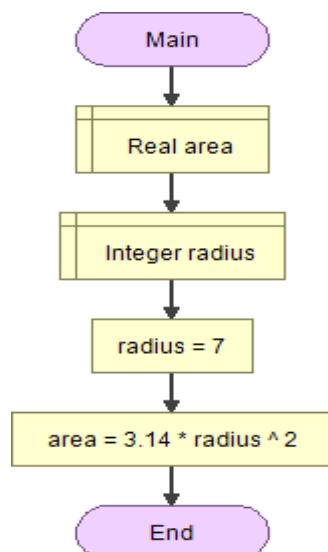


Cosa fa

Il comando di dichiarazione viene usato per creare variabili e vettori (arrays). Sono utilizzati per conservare i dati nel frattempo che il programma è in esecuzione.

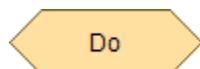
Esempio

Nel seguente esempio vengono dichiarate due variabili: "area" (che conterrà numeri reali) e "radius" (che conterrà numeri interi). Viene poi utilizzato il comando di assegnazione per impostare a 7 la variabile 'radius'. Per finire, nella variabile "area" verrà assegnato il calcolo dell'area di un cerchio.



## Do

Figura associata

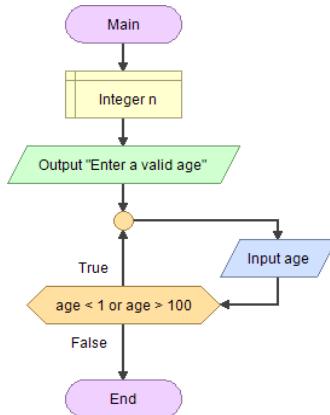


Cosa fa

Un ciclo Do è simile al ciclo While tranne per il fatto che le istruzioni al suo interno vengono eseguite almeno una volta prima di poterne verificare la condizione di ciclo.

Esempio

Il seguente esempio mostra il ciclo DO che accetta solo un input valido, una età compresa tra 1 e 100. Se age < 1 o age > 100 ripete il ciclo.



## For

Figura associata

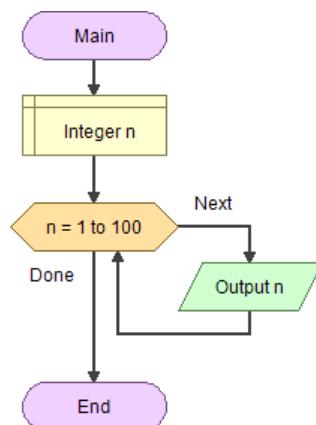


Cosa fa

Il ciclo enumerativo For incrementa una variabile all'interno di un intervallo di valori assegnati. Questa struttura, di solito, rimpiazza la struttura While.

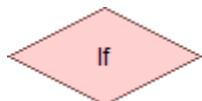
Esempio

Il seguente esempio visualizza a monitor i numeri da 1 a 100.



## If

Figura associata



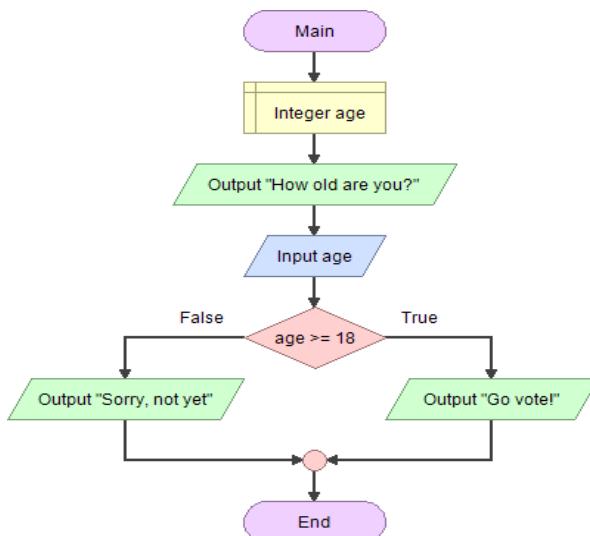
Cosa fa

Il comando If controlla una espressione booleana e passa al ramo del Vero o del Falso a seconda se è verificata o meno la condizione.

Esempio

Il seguente esempio dichiara una variabile di tipo intero chiamata 'age'. Viene inserita, quindi, l'età da tastiera per controllare se l'utente può andare a votare o meno.

Alla fine, una struttura If controlla lse l'età inserita da tastiera e assegnata alla variabile "age" è maggiore o uguale a 18. Se si è maggiorenne. Il programma visualizzerà la frase "Go vote!" altrimenti visualizzerà la frase "Sorry, not yet".



## Input

Figura associata

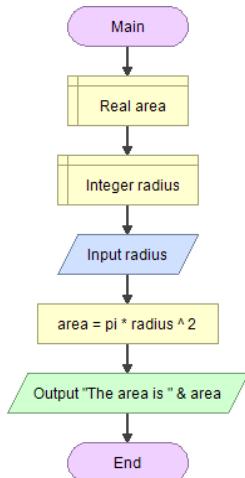


Cosa fa

Un Input legge un valore da tastiera e immagazzina tale valore in una variabile.

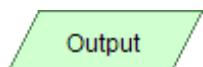
## Esempio

Il seguente esempio crea due variabili: 'area' e 'radius'. Utilizza poi un Input per leggere il raggio da tastiera e assegnarlo alla variabile 'radius'. Alla fine verrà visualizzato un Output sullo schermo del risultato del calcolo dell'area.



## Output

Figura associata

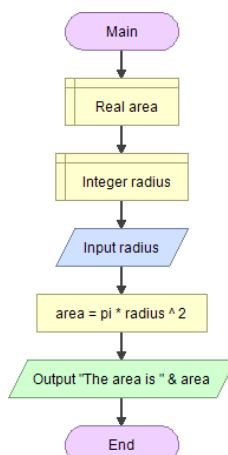


Cosa fa

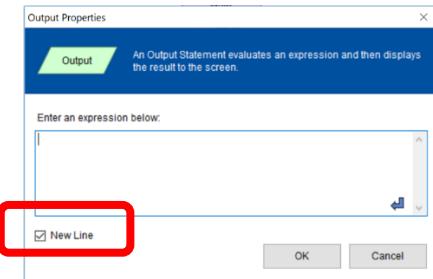
Un comando di Output valuta un'espressione e poi visualizza il risultato sullo schermo.

## Esempio

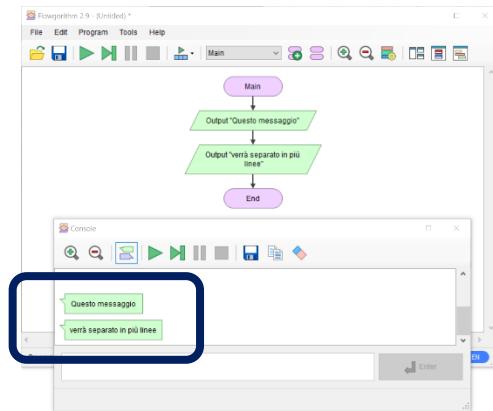
Il seguente esempio crea due variabili: 'area' e 'radius'. Utilizza poi un Input per leggere il raggio da tastiera e assegnarlo alla variabile 'radius'. Alla fine verrà visualizzato un Output del risultato del calcolo dell'area.



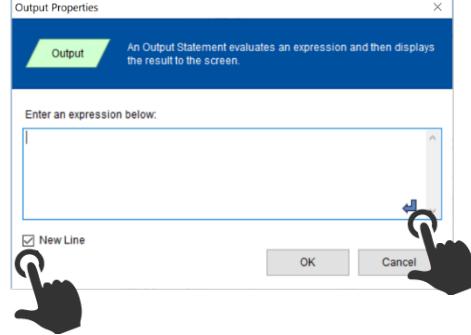
Dalla versione 2.9 è stata inserita, per l'output, la possibilità di utilizzare la funzione "Ritorno a capo" (in inglese: "New Line")



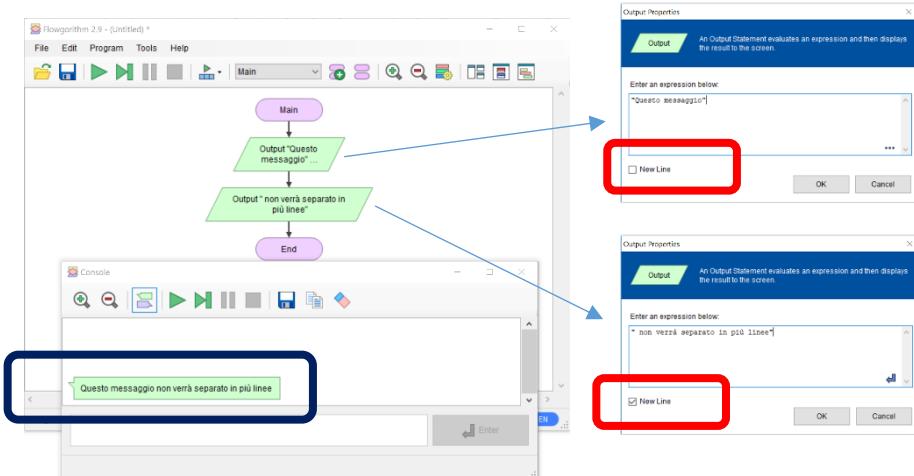
Questa opzione influisce sulla visualizzazione dei messaggi con o senza il ritorno a capo. Se utilizzata Flowgorithm creerà un messaggio diverso per ogni istruzione di Output utilizzata.



Se invece non si imposta questa opzione, cioè si toglie la spunta all'opzione, cliccando col mouse come da figura



si otterrà un unico messaggio con ritorno a capo nell'ultima parte del messaggio:



## While

Figura associata

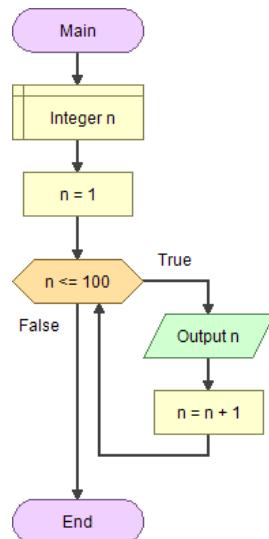


Cosa fa

Il ciclo While valuta una espressione booleana e, se vera, esegue le istruzioni contenute al suo interno. Cicla fino a quando la condizione diventa falsa.

Esempio

Il seguente esempio visualizza i numeri da 1 a 100. Il comando di assegnazione "n = n + 1" incrementa la variabile 'n' ogni iterazione del ciclo.



# Grafica Tartaruga

## Ruotare

Figura associata

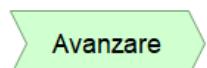


Cosa fa

Cambia l'angolo ruotando la Tartaruga a sinistra o a destra di un numero specificato di gradi.

## Avanzare

Figura associata

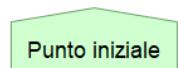


Cosa fa

Sposta la tartaruga della distanza specificata, con pennino giù o pennino sù

## Punto iniziale

Figura associata



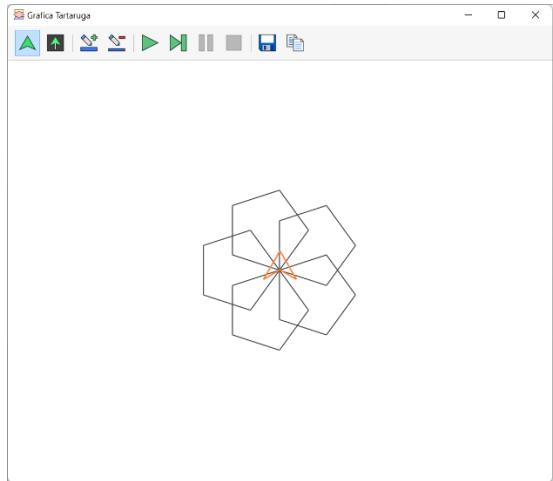
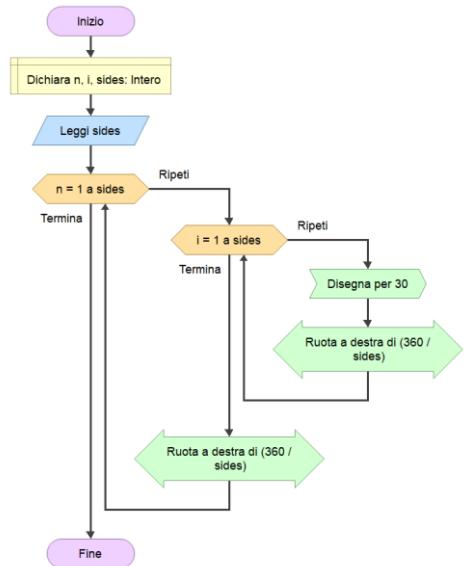
Cosa fa

Riporta la tartaruga alla posizione iniziale e imposta l'angolo a 90 gradi.

## Esempi

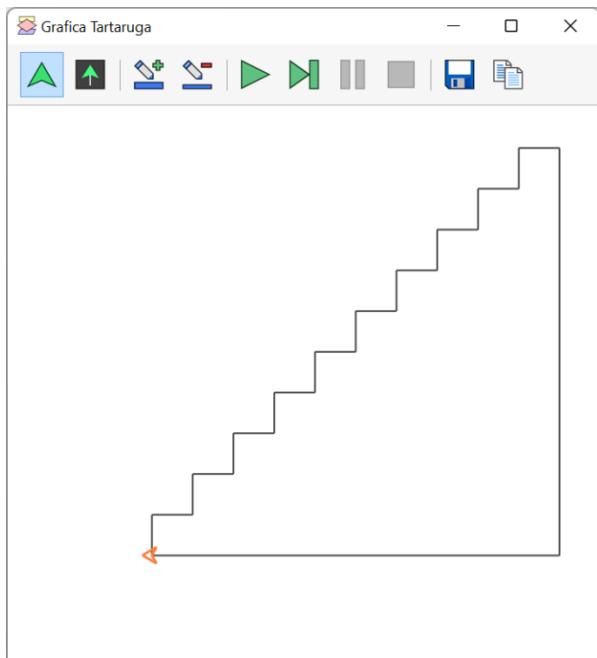
Disegno di un fiore

Con un input di 5 avremo ....



Scalinate

Con un input di 10 avremo....

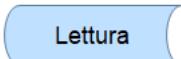


# Gestione File

La versione 3 supporta la lettura e la scrittura di file di testo. Per supportare questa funzione, vengono introdotte quattro nuove figure.

## Lettura

Figura associata

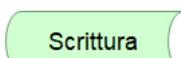


Cosa fa

Un'istruzione di Lettura legge un valore dal file aperto e lo memorizza in una variabile.

## Scrittura

Figura associata



Cosa fa

Un'istruzione di Scrittura valuta un'espressione e scrive il risultato nel file aperto.

## Chiusura

Figura associata

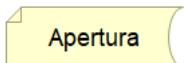


Cosa fa

Un'istruzione di Chiusura chiude un file aperto.

## Apertura

Figura associata

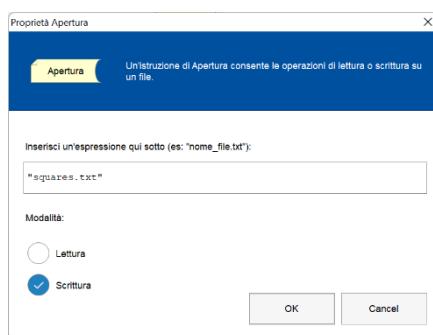


Cosa fa

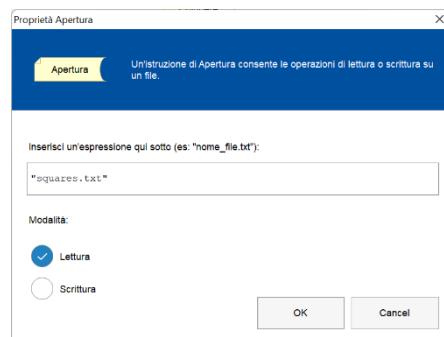
Un'istruzione di Apertura consente le operazioni di lettura o scrittura su un file.

Esempio

↓  
Aprire "squares.txt" in  
scrittura

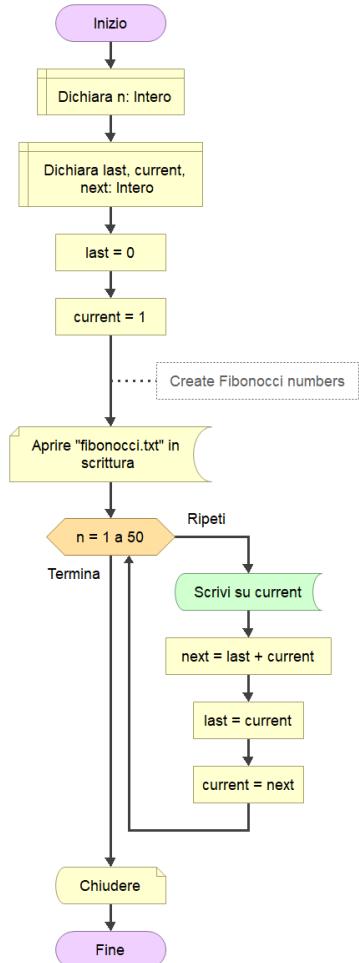


↓  
Aprire "squares.txt" in lettura



## Esempi

### Fibonacci con utilizzo di file

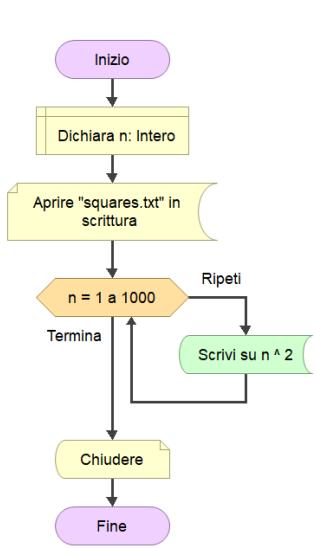


fibonacci.txt - Blocco note

File	Modifica	Visualizza
1		
1		
2		
3		
5		
8		
13		
21		
34		
55		
89		
144		
233		
377		
610		
987		
1597		
2584		
4181		
6765		
10946		
17711		
28657		
46368		
75025		
121393		
196418		
317811		
514229		
832040		
1346269		
2178389		
3524578		
5702887		
9227465		
14930352		
24157817		
39088169		
63245986		
102334155		
165580141		

Linea 1, colonna 1

### Calcolo dei quadrati



squares.txt - Blocco note

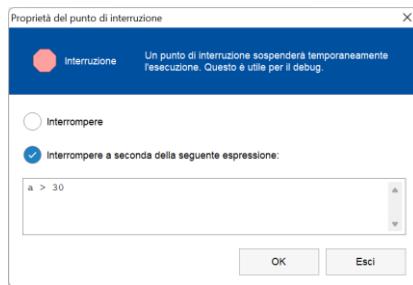
File	Modifica	Visualizza
1		
4		
9		
16		
25		
36		
49		
64		
81		
100		
121		
144		
169		
196		
225		
256		
289		
324		
361		
400		
441		
484		
529		
576		
625		
676		
729		
784		
841		
900		
961		
1024		
1089		
1156		
1225		
1296		

# Interruzioni condizionali

La versione 3 consentirà agli studenti di creare "Interruzioni condizionali".

Attualmente, la figura del punto di interruzione  Interruzione metterà sempre in pausa il programma. La nuova versione può utilizzare questo comportamento (l'impostazione predefinita) o interrompere solo se un'espressione condizionale è vera.

Questo dovrebbe essere un po' più utile, specialmente per le dimostrazioni in classe.



# Per gli Amministratori di Sistema



Come Flowgorithm utilizza il file registro di Windows

Flowgorithm usa il file registro di Windows per conservare le impostazioni utente. Sono conservative solo poche impostazioni. Queste includono linguaggio, colori, modello di diagramma e massima grandezza dello stack interno di Flowgorithm.

Quando si esegue Flowgorithm, verrà controllata prima la chiave CURRENT\_USER per caricare i valori in essa contenuti. Se non la trova, verrà controllata la chiave LOCAL\_MACHINE.

Se l'applicazione viene eseguita in modalità Amministratore, Flowgorithm salverà le sue impostazioni nella chiave di registro LOCAL\_MACHINE. Altrimenti la salverà in CURRENT\_USER. Se si vuole impostare i settaggi di default per tutti gli utenti del laboratorio, apri l'applicazione in modalità Amministratore e imposta linguaggio, colore, ecc... Ogni nuovo utente (che non possiede impostazioni utente), userà queste.

The screenshot shows the Windows Registry Editor window. The left pane displays a tree view of registry keys under 'HKEY\_LOCAL\_MACHINE'. The 'SOFTWARE' key is expanded, showing various subkeys like {46577E3C-95B4-4f4f-}, 7-Zip, AMD, Apple Inc., ATI, ATI Technologies, AVG, AVG Secure Search, Classes, Clients, CyberLink, Dell, and Flowgorithm. The 'Flowgorithm' key is selected and highlighted with a blue border. The right pane is a table showing the values of the 'Flowgorithm' key:

Name	Type	Data
(Default)	REG_SZ	(value not set)
Chart Style	REG_SZ	Default
Color Scheme	REG_SZ	Pastel
Language	REG_SZ	en
Max Stack Size	REG_SZ	5000

The status bar at the bottom of the window shows the path: Computer\HKEY\_LOCAL\_MACHINE\SOFTWARE\Flowgorithm.

## Chiavi Registro

Chiave	Descrizione
Chart Style	Contiene il nome del modello ("Default", "Classic", ecc...) Se non viene riconosciuto il valore della chiave, l'applicazione utilizzerà il modello "Default".

Color Scheme	Contiene il nome dello schema colori corrente ("Pastel", "Cyberspace", ecc...) Se non viene riconosciuto il valore della chiave, l'applicazione utilizzerà il schema colori "Pastel".
Language	Contirnr il codice ISO del linguaggio visualizzato. Se non viene riconosciuto il valore della chiave, l'applicazione utilizzerà il codice lingua "en" (English).
Max Stack Size	Contiene la grandezza massima dello stack. Se l'applicazione eccede questa grandezza, ad esempio nel caso di una chiamata ricorsiva infinita, verrà visualizzata l'informazione "Stack Overflow Error". Questa impostazione non può essere modificata dalla applicazione.

Flowgorithm utilizza InstallShield. Se si desidera utilizzare un'installazione invisibile all'utente (ad esempio, i computer di laboratorio), utilizzare i seguenti argomenti. NOTA: la spaziatura è significativa per l'opzione / v.

```
Setup /s /v" /qn"
```

La versione corrente di Flowgorithm utilizza Microsoft Installer Se si desidera utilizzare un'installazione invisibile all'utente (ad esempio, i computer di laboratorio), utilizzare i seguenti argomenti.

```
Setup /quiet
```

## Chiavi utente

Le seguenti chiavi verranno applicate solo a CURRENT\_USER. Anche se l'applicazione è aperta in modalità amministratore, questi verranno salvati nell'hive utente corrente.

HKEY_CURRENT_USER\Software\Flowgorithm			
Key	Default	App Can Change?	Description
MRU 0	Null string	Yes	Most recently used file list.
MRU 1	Null string	Yes	Most recently used file list
MRU 2	Null string	Yes	Most recently used file list
MRU 3	Null string	Yes	Most recently used file list
MRU 4	Null string	Yes	Most recently used file list

## Stampante

Nella versione 2.6.0 e successive, Flowgorithm salva le informazioni della stampante nel registro.

<Hive>\Software\Flowgorithm\Printer			
Key	Default	App Can Change?	Description
Landscape	False	Yes	Boolean value that specifies if the printer will default ot landscape (true) or portrait (false).
Margin Bottom	0.75	Yes	Margin in inches.
Margin Left	0.75	Yes	Margin in inches.
Margin Right	0.75	Yes	Margin in inches.
Margin Top	0.75	Yes	Margin in inches.
Name		Yes	Name of the last printer used.

## Colori

Schemi di colori personalizzabili sono stati aggiunti nella versione 2.5. Se l'utente carica un file di combinazioni di colori, il tasto "Schema colori" è impostato su "Personalizzato" e i colori vengono memorizzati nella sottocartella "Colori" di Flowgorithm. Se il "Combinazione colori" principale è impostato su qualcosa' altro, questi tasti vengono ignorati.

<Hive>\Software\Flowgorithm\Colors			
Key	Default	App Can Change?	Description
Name	n/a	Yes	Name of the current custom color scheme. This is displayed on dropdown controls
Assign	n/a	Yes	Assignment shape colors
Breakpoint	n/a	Yes	Breakpoint shape colors
Call	n/a	Yes	Call shape colors
Comment	n/a	Yes	Comment shape colors
Declare	n/a	Yes	Declare shape colors
Error	n/a	Yes	Shape error colors
Executing	n/a	Yes	Shape executing colors
Header	n/a	Yes	Add Shape Menu and Line Numbers (source code viewer)
Hover	n/a	Yes	Mouseover effect in the Add Shape Menu
If	n/a	Yes	If shape colors
Incomplete	n/a	Yes	Shape incomplete colors
Input	n/a	Yes	Input shape colors
Line	n/a	Yes	Line colors
Loop	n/a	Yes	Loop shape (For, While, Do) colors
Output	n/a	Yes	Output shape colors
Page	n/a	Yes	The background colors
Selected	n/a	Yes	Shape selection colors
Terminator	n/a	Yes	Terminator shape colors
Window	n/a	Yes	Edit window background colors

## Visualizzatore codice sorgente

Nella versione 2.12.0 e successive, Flowgorithm salva le chiavi relative alla lingua e ai numeri di riga correnti

<Hive>\Software\Flowgorithm\Source Code Viewer			
Key	Default	App Can Change?	Description
Language	Auto	Yes	The name of the current programming language that the Source Code Viewer will use to generate code.
Line Numbers	false	Yes	If the window displays line numbers

# Contatto

e-mail: [Flowgorithm @ DevinCook.com](mailto:Flowgorithm@DevinCook.com).

# Informazioni

## Origine del Nome

Il nome è la fusione dei termini "flowchart" e "algorithm".

## Scritto in

Flowgorithm è implementato utilizzando Microsoft C#.

## Autore

Devin Cook

# Licenza



# Ulteriori Informazioni

- [Older Revisions](#)