

AlgoBuild 0.75 - Manuale minimo

Sommario

- [Cos'è AlgoBuild](#)
 - [Novità rispetto alle versioni precedenti](#)
 - [Formato eseguibile e requisiti di sistema](#)
 - [Installare il programma](#)
 - [Avviare il programma](#)
 - [Inserire una istruzione](#)
 - [Modificare una istruzione](#)
 - [Copiare/tagliare una istruzione](#)
 - [Incollare una istruzione dagli appunti](#)
 - [Salvare il lavoro](#)
 - [Aprire un lavoro dal disco](#)
 - [Stampare un diagramma o esportare in formato immagine](#)
 - [Eseguire un programma passo-passo](#)
 - [Selezionare font, dimensione caratteri, spessore della linea](#)
 - [Sintassi](#)
-

Cos'è AlgoBuild

AlgoBuild è un software didattico per progettare programmi. Presenta un ambiente visuale in cui disegnare dei flow-chart rispettando le regole della programmazione strutturata. Il programma prodotto viene visualizzato anche in forma di pseudo-codice.

Se le istruzioni vengono inserite rispettando una opportuna sintassi è possibile eseguire il programma passo-passo.

Lo stadio di sviluppo del software è, per ora, solo a livello di prototipo ed ha quindi possibilità ancora piuttosto limitate.

[Vai al Sommario](#)

Novità rispetto alla versione 0.73

Differenze rispetto alla versione precedente:

- Aggiunta la gestione degli array (vettori) monodimensionali
- Scrolling automatico della finestra del flow chart durante l'esecuzione
- Spostamento della selezione dei caratteri/spessore linea in una finestrella apposita accessibile con il pulsante "a" della barra strumenti
- Miglioramento dell'allineamento delle scritte negli elementi grafici
- BUGFIX: Aggiunta l'estensione di default ".bfc" nel salvataggio/apertura file

Novità rispetto alla versione 0.72

Differenze rispetto alla versione precedente:

- Selezione Tipo Font
- Scelta dimensione font per codice e per output
- Seleziona spessore linea del grafico

Novità rispetto alla versione 0.71

Vaersione attuale: AlgoBuild 0.72.

Ultimo aggiornamento 27/04/2013

1 - Numeri pseudo casuali

È stata aggiunta la funzione random per la generazione di numeri pseudo casuali. La nuova funzione non ha argomenti e restituisce un numero maggiore o uguale a 0.0 e minore di 1.0.

Esempi:

`num=random()` assegna un numero pseudo casuale tra 0.0 e 0.999999999999999 alla variabile num.
`dado=floor(random()*6+1)` assegna uno dei sei valori a caso tra 1.0, 2.0, 3.0, 4.0, 5.0 e 6.0 alla variabile dado.

2 - Operazioni di input/output e trace

Nelle operazioni di I/O da ora viene omessa la descrizione "INPUT" e "OUTPUT" se il Trace è disabilitato.

Esempio con trace abilitato:

```
OUTPUT i: 0.0
OUTPUT i: 1.0
OUTPUT i: 2.0
ecc...
```

Esempio con trace disabilitato:

```
0.0
1.0
2.0
ecc...
```

3 - Help online / Facebook / YouTube

Sono state aggiunte altre voci al menu Aiuto:

- Collegamento all'aiuto online sul sito algotobuild.com
- Collegamento alla pagina FaceBook di Algotobuild
- Collegamento al canale YouTube di Algotobuild

4 - BUGFIX - operatore "NOT"

Nella versione 0.71 era presente un errore di programmazione e l'operatore logico not (rappresentato con il punto esclamativo prefisso !) non veniva valutato correttamente. In questa versione l'errore è stato eliminato e il NOT logico dà i risultati aspettati.

Novità rispetto alla versione 0.7

1 - Copia immagine

Con questo comando l'immagine del diagramma viene copiata negli appunti e può essere incollata all'interno di programmi di grafica (Paint, Gimp, Photoshop ecc...), all'interno di documenti tipo office o nel notepad della LIM.

2 - BUGFIX - operatore "or"

Nella versione 0.7 era presente un errore di programmazione e l'operatore logico or (rappresentato con due sbarrette verticali ||) non veniva valutato correttamente. In questa versione l'errore è stato eliminato e l'OR logico dà i risultati aspettati.

3 - Informazioni su...

Dalla finestra di messaggio "Informazioni su..." è possibile accedere al sito algotool.com premendo sul bottone corrispondente.

Novità rispetto alla versione 0.6

1 - Splash screen

Adesso all'avviamento il programma si presenta.

Lo splash screen è una finestra o immagine visualizzata all'avvio di un programma.

Quello di AlgoBuild riporta un messaggio di saluto (*), nome e numero di versione; è simile a quello del sito internet ufficiale di AlgoBuild.

[Vai al sito \[algotool.com\]\(http://algotool.com\)](http://algotool.com)

(*) "Hello World!" è tradizionalmente il messaggio visualizzato dal primo programma quando si impara un nuovo linguaggio.

2 - Esecuzione temporizzata

Nella barra degli strumenti sono stati introdotti tre elementi nuovi: una casella di spunta (checkbox), una casella numerica e il pulsante Pausa Esecuzione.

La casella di spunta, normalmente selezionata, riporta la dicitura "Passo passo". Quando è selezionata l'esecuzione avviene un passo alla volta come sempre. Se viene deselezionata,

premendo il pulsante "Esegui istruzioni" con la freccia verde, le istruzioni verranno eseguite una di seguito all'altra con la temporizzazione stabilita dalla casella numerica "Tempo (100-5000 ms)".

Come indicato il tempo tra una istruzione e la successiva va da un minimo di 100 ms (pari a 0,1 s) a un massimo di 5000 ms (pari a 5 s). I tempi sono puramente indicativi.

La pressione del pulsante "Pausa Esecuzione" fa arrestare il programma. Premendo "Esegui istruzioni" l'esecuzione riprende da dove è stata interrotta.

La modifica del tempo è recepita solo se il programma è in pausa o interrotto.

3 - Identificatori "case-sensitive"

Da questa versione c'è differenza tra maiuscolo e minuscolo, quindi x e X sono due variabili diverse.

4 - Risoluzione di un errore relativo all'interpretazione delle espressioni

Se una qualsiasi variabile aveva all'interno una parola chiave avveniva un errore.

È capitato con "distanza". Il programma si bloccava inspiegabilmente. Ciò era dovuto alla presenza del nome di funzione "tan" all'interno della variabile "distanza".

5 - Eseguitibile ".exe" per Windows

Per semplificare l'utilizzo del programma, questo adesso viene distribuito anche in formato

eseguibile per windows.

NOTA: il runtime Java JRE 6 (o successivo) è ancora necessario.

[Vai al Sommario](#)

Formato eseguibile e requisiti di sistema

Il programma è distribuito in due versioni: come archivio eseguibile Java e come programma eseguibile Windows.

Per poter essere utilizzato è necessario aver installato nel computer un ambiente di esecuzione Java (JRE) versione 6.

Siccome il programma è basato su oggetti e librerie Java standard è eseguibile su qualsiasi piattaforma che supporti Java 6.

La versione in formato jar è utilizzabile su Windows, Linux, Mac e qualsiasi altra piattaforma mentre la versione in formato exe è utilizzabile solo in Windows.

[Vai al Sommario](#)

Installare il programma

1 - Scompattare l'archivio zip (cartella compressa)

2 - Copiare il file AlgoBuild071.jar o AlgoBuild071.exe in una cartella.
Per comodità è opportuno che sia la cartella di lavoro.

[Vai al Sommario](#)

Avviare il programma

Il file viene distribuito con il nome AlgoBuild071.jar o AlgoBuild071.exe.

La versione eseguibile Windows parte con un doppio click del mouse.

Per la versione jar il discorso è più articolato e si possono presentare tre possibilità

1. **Doppio clic:**

È il caso (abbastanza comune) più semplice; nel sistema i file jar sono associati al runtime Java. Se questo modo non funziona passare al modo 2.

2. **Avvio da console:**

È il caso (meno comune) sfortunato a livello 1; nel sistema i file jar NON sono associati al runtime Java.

Per avviare il programma bisogna aprire un terminale (in Windows: Esegui -> cmd), spostarsi nella cartella contenente il programma e invocare java direttamente con il comando: `java -jar AlgoBuild071.jar`

In questo caso è più opportuno aggiungere un collegamento o un file batch (shell script per i sistemi Unix/Linux) per semplificare l'operazione di avvio. Se anche questo modo non funziona passare al modo 3.

3. **Avvio da console con path completo:**

È il caso (ancora meno comune) sfortunato a livello 2; nel sistema i file jar NON sono associati al runtime Java e il programma java NON è nel PATH di ricerca del sistema operativo.

Per avviare il programma bisogna aprire un terminale (in Windows: Esegui -> cmd), spostarsi nella cartella contenente il programma e invocare java direttamente specificando tutto il path: `C:\Programmi\Java\jre6\bin\java -jar AlgoBuild071.jar`

Naturalmente il percorso `C:\Programmi\Java\jre6\bin\` dipende da dove è stato installato jre nel sistema in questione.

Anche in questo caso è più opportuno aggiungere un collegamento o un file batch (shell script per i sistemi Unix/Linux) per semplificare l'operazione di avvio.

Se il programma si è avviato regolarmente comparirà la finestra principale simile a questa:



[Vai al Sommario](#)

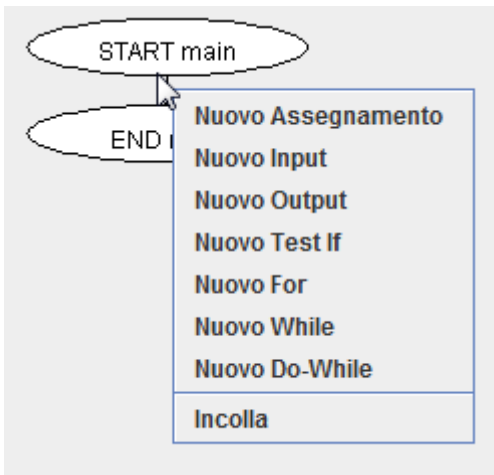
Inserire una istruzione

Tra le ellissi Start e End si trova una freccia verso il basso.

Spostandovi il mouse sopra la freccia viene sostituita da un quadrato rosso.



Premendo il pulsante sinistro del mouse compare un menu da cui è possibile scegliere l'istruzione da inserire.



Nota: ogni freccia verso il basso è un punto in cui è possibile inserire una istruzione.

[Vai al Sommario](#)

Modificare una istruzione

Spostare il puntatore del mouse sopra l'istruzione da modificare, premere il pulsante sinistro del mouse e scegliere Modifica dal menu.

[Vai al Sommario](#)

Copiare/tagliare una istruzione

Spostare il puntatore del mouse sopra l'istruzione da modificare, premere il pulsante sinistro del mouse e scegliere Copia o Taglia dal menu.

Effetti:

- Copia invia negli appunti una copia in formato testuale dell'istruzione scelta (è possibile inserirla in qualsiasi editor di testo...).
- Taglia invia negli appunti una copia in formato testuale dell'istruzione scelta ed elimina l'istruzione stessa dal punto in cui si trovava.

[Vai al Sommario](#)

Incollare una istruzione dagli appunti

Spostare il puntatore del mouse sopra a un punto di inserimento (freccia verso il basso) premere il pulsante sinistro del mouse e scegliere Incolla dal menu.

[Vai al Sommario](#)

Salvare il lavoro

Dal menu principale scegliere File -> Salva... oppure File -> Salva con nome...

In alternativa è possibile schiacciare uno dei pulsanti con il floppy disk sulla barra degli strumenti.

[Vai al Sommario](#)

Aprire un lavoro dal disco

Dal menu principale scegliere File -> Apri...

In alternativa è possibile schiacciare il pulsante con la cartellina sulla barra degli strumenti.

[Vai al Sommario](#)

Stampare un diagramma o esportare in formato immagine

Dal menu principale scegliere File -> Stampa oppure File -> Esporta immagine

[Vai al Sommario](#)

Selezionare font, dimensione caratteri, spessore della linea

Sulla barra degli strumenti schiacciare il bottone con l'icona "a".

Comparirà una finestrella da cui è possibile scegliere:

- Nome/tipo del carattere da usare
- Dimensioni caratteri della vista "codice"
- Dimensioni caratteri della finestra output e della zona delle variabili
- Spessore della linea usata per il disegno

[Vai al Sommario](#)

Eseguire un programma passo-passo

NOTA: per eseguire un programma in modalità passo passo, cioè una istruzione alla volta, è necessario che la casella "Passo passo" sia selezionata.

Premere il pulsante con la freccia verde sulla barra degli strumenti.

Ad ogni pressione l'esecuzione del programma avanza di una istruzione.

Alcune istruzioni (test e cicli) necessitano anche di più pressioni in quanto deve essere anche valutata la condizione o incrementate le variabili (for).

La casella di spunta "Traccia" riporta in output ogni messaggio e lo stato delle variabili a ogni istruzione.

Togliendo la spunta vengono visualizzati solo input e output.

Per interrompere l'esecuzione del programma prima di arrivare a END premere il pulsante con il quadrato rosso.

Se non si interrompe l'esecuzione non è possibile modificare il programma.

NOTE

Se durante l'esecuzione vengono trovati degli errori (es. espressioni errate) il programma termina riportando un messaggio di errore.

[Vai al Sommario](#)

Eseguire un programma in modalità temporizzata

NOTA: per eseguire un programma in modalità temporizzata, cioè una istruzione dietro l'altra a intervalli di tempo regolari, è necessario che la casella "Passo passo" sia deselezionata.

Premere il pulsante con la freccia verde sulla barra degli strumenti.

Il programma si avvia e avanza di una istruzione alla volta a intervalli regolari. Alcune istruzioni (test e cicli) necessitano anche di più tempo in quanto deve essere anche valutata la condizione o incrementate le variabili (for).

Se durante l'esecuzione si preme il pulsante "Pausa Esecuzione" il programma viene fermato momentaneamente. Premendo "Esegui Istruzioni" riprende da dove era stato interrotto. La modifica del tempo è recepita solo se il programma è in pausa o interrotto.

La casella di spunta "Traccia" riporta in output ogni messaggio e lo stato delle variabili a ogni istruzione.

Togliendo la spunta vengono visualizzati solo input e output.

Per interrompere l'esecuzione del programma prima di arrivare a END premere il pulsante con il quadrato rosso.

Se non si interrompe l'esecuzione non è possibile modificare il programma.

NOTE

Se durante l'esecuzione vengono trovati degli errori (es. espressioni errate) il programma termina riportando un messaggio di errore.

[Vai al Sommario](#)

Sintassi

Si riporta in breve le principali regole e la sintassi delle istruzioni.

- Nomi di variabili: stringa in minuscolo, MAIUSCOLO o miStO. Può contenere un numero.
Attenzione: C'È DIFFERENZA tra Maiuscolo e minuscolo.
Per mantenere le usanze in voga si consiglia di usare nomi minuscoli.
- Array (vettori): Si utilizzano come i nomi di variabili seguiti da un indice (numero o espressione) tra una coppia di parentesi quadre.
Esempio: $a[0]=7$ oppure $b[i]=a*3$ oppure $x=v1[i]$
NOTA BENE: L'indice è un valore numerico che viene interpretato come numero intero eventualmente troncato. L'indice minimo è 0.
- Tipi di dati: le variabili sono **tutte di tipo double** (decimali a doppia precisione).
- Assegnamento:
 $variabile=espressione$ oppure
 $vettore[indice]=espressione$
Esempio: $x=7$ oppure $x=x+1$ oppure $giorni[mese]=31$
Nota: NON si possono inserire più istruzioni in un blocco.
- Input:
 $variabile$ oppure
 $vettore (array)$
Nota: è possibile leggere una sola variabile in ogni blocco.
- Output:
 $variabile$ oppure
 $vettore (array)$ oppure

<="" i=""> oppure
"messaggio"

Nota: è possibile visualizzare una sola variabile, array, espressione o messaggio in ogni blocco.

- If, While e Do-while:
espressione booleana
Esempio: $y < 5$ oppure $x = z * 2$ oppure $a[i] < a[i+1]$
- For:
assegnamento; espressione booleana; assegnamento
Esempio: $i = 0; i < 5; i = i + 1$
- Espressioni:
 - Operatori aritmetici
 - + addizione
 - sottrazione
 - * moltiplicazione
 - / divisione
 - % modulo (resto della divisione)
 - ^ potenza (es. x^2)
 - Funzioni matematiche
 - sqrt(...) radice quadrata
 - sin(...) funzione trigonometrica
 - cos(...) idem
 - tan(...) idem
 - cotan(...) idem
 - asin(...) funzioni trigonometriche inverse
 - acos(...) idem
 - atan(...) idem
 - acotan(...) idem
 - sinh(...) funzione iperbolica
 - cosh(...) idem
 - tanh(...) idem
 - exp(...) e elevato alla ...
 - ln(...) logaritmo naturale
 - abs(...) valore assoluto
 - ceil(...) arrotondamento intero per eccesso
 - floor(...) arrotondamento intero per difetto
 - round(...) arrotondamento all'intero più vicino
 - random() numero casuale tra 0.0 e 0.9999999999999999
 - Operatori logici:
 - ==
 - !=
 - && And logico
 - || Or logico
 - >
 - <
 - >=
 - <=
 - !
 - Costanti e valori particolari: (non possono essere usati come nomi di variabili)
 - PI pigreco
 - EULER numero di eulero

INFINITY più infinito
-INFINITY meno infinito
NaN not a number (0/0)
TRUE equivale a 1.0
FALSE equivale a 0.0