

FLOWGORITHM

Dal Diagramma di Flusso all'Algoritmo

Laboratorio Informatica — Classe Prima

Nome: _____ Classe: _____

Modulo 01 — Installazione e Primo Programma

Scarica Flowgorithm, esplora l'interfaccia, crea il tuo primo diagramma.

1. Scaricare e installare

Vai su flowgorithm.org → Download → scegli la versione per il tuo sistema operativo. Avvia il file e segui l'installazione guidata.

2. Main e End

Ogni diagramma parte da due terminali: "Main" (non "Start") e "End". Tutto il programma va inserito sulla linea tra i due.

3. Aggiungere un blocco

Passa il mouse sulla linea tra Main e End finché diventa arancione, poi doppio clic (o clic destro) per aprire il menu delle forme disponibili.

4. Il blocco Output

Clicca sul parallelogramma verde Output. Apparirà grigio (incompleto). Doppio clic per aprire "Output Properties" e scrivi, tra virgolette:

```
"Hello, world!"
```

Il blocco diventerà verde: è completo.

5. Eseguire il programma

Clicca sul Play verde, oppure premi F5, oppure Program → Run. Si apre la Console Screen con il risultato.

Bonus: Flowgorithm può tradurre il diagramma in codice vero (C, Java, Python...) — cerca "Generate Source Code" nel menu.

Modulo 02 — Analisi del Problema

Il modello IPO, i simboli di Flowgorithm, come tracciare un algoritmo a mano.

1. Il modello IPO

Ogni problema si scompone in: Input (cosa entra), Processing (cosa si calcola), Output (cosa esce).

Esempio — "Calcola l'area di un rettangolo": *Input* = base, altezza; *Processing* = area = base × altezza; *Output* = area.

2. Simboli principali

Simbolo	Significato
Terminale (Main/End)	Inizio e fine del programma
Input / Output	Chiede o mostra un dato (parallelogramma)
Dichiarazione / Assegnazione	Crea una variabile o le assegna un valore
Se (If)	Decisione, crea due percorsi (diamante)
Mentre / Per (While/For)	Ripete un gruppo di blocchi

3. Tracciare un diagramma a mano

Esempio svolto — Diagramma: Input numero → doppio = numero × 2 → Output doppio. Se l'utente inserisce 7, la tabella si compila così (il trattino "—" indica un valore non ancora calcolato):

Passo	numero	doppio	Output
Input numero	7	—	—
doppio = numero × 2	7	14	—
Output doppio	7	14	14

Prova tu — stesso procedimento, ma con: Input numero → triplo = numero × 3 → Output triplo. Se l'utente inserisce 5, completa tu la tabella:

Passo	numero	triplo	Output
Input numero			
triplo = numero × 3			
Output triplo			

Modulo 03 — Variabili, Input e Output

Dichiarare variabili, chiedere dati, mostrare risultati.

1. Tipi di dato

Tipo	Esempio
Integer	Numeri interi (25, -3, 0)
Real	Numeri con virgola (3.14)
String	Testo ("Mario")
Boolean	Vero o Falso

2. Input e Output

Input legge un valore in una variabile GIÀ dichiarata. Output mostra testo, variabili, o entrambi uniti con &:

```
Output "Hai " & eta & " anni"
```

Esercizio: Il Biglietto da Visita Digitale

1. Dichiarare nome (String) ed eta (Integer). 2. Chiedi "Come ti chiami?" e leggi nome. 3. Chiedi "Quanti anni hai?" e leggi eta. 4. Mostra: "Ciao " & nome & "! Tra " & (100-eta) & " anni compirai 100 anni."

Modulo 04 — Operatori e Sequenza

Calcoli, l'ordine delle istruzioni, i primi algoritmi utili.

1. Operatori aritmetici

Operatore	Significato
+ - * /	Addizione, sottrazione, moltiplicazione, divisione
Mod	Resto della divisione (5 Mod 2 → 1)
^	Potenza (2 ^ 3 → 8)
&	Unisce testo e variabili in un Output

2. La sequenza: l'ordine conta

Un diagramma senza decisioni o cicli è una sequenza: i blocchi eseguono uno dopo l'altro. Scambiare l'ordine di due blocchi può cambiare il risultato.

Esercizio 1: Splitta il conto

1. Dichiarare e chiedere totaleConto (Real), numeroAmici (Integer). 2. $quotaPerPersona = totaleConto / numeroAmici$. 3. Mostra il risultato.

Esercizio 2: Convertitore di Valuta

1. Dichiarare e chiedere euro (Real), tassoDiCambio (Real). 2. $dollari = euro * tassoDiCambio$. 3. Mostra il risultato.

Modulo 05 — Strutture di Selezione

Prendere decisioni: Se, Se-Altrimenti, alternative annidate.

1. Il blocco Se (If)

Ha la forma di un rombo: contiene una condizione. Se è vera, un percorso; se falsa, l'altro.

2. Operatori di confronto e logici

Operatore	Significato
> < >= <=	Maggiore, minore, maggiore/minore o uguale
== <>	Uguale, diverso
And / Or / Not	Combinano più condizioni
Flowgorithm accetta sia = che == per l'uguaglianza in una condizione: da qui in poi usiamo sempre ==, come in Python/Java/JavaScript. L'= singolo resta invece per l'assegnazione (es. totale = 10).	

3. Più di due alternative: Se annidati

Flowgorithm non ha un blocco Switch/Case: per più di due alternative si annidano più blocchi Se, uno dentro il ramo Falso del precedente.

Esercizio 1: Il buttafuori della discoteca

Chiedi eta. Se eta<14 → "Solo pomeridiano". Altrimenti se eta<18 → "Chiedi ai genitori". Altrimenti → "Puoi entrare!".

Esercizio 2: Che sconto hai?

Chiedi spesa. Se spesa>=100 → sconto 20%. Altrimenti se spesa>=50 → sconto 10%. Altrimenti → nessuno sconto. Mostra la spesa finale.

Modulo 06 — Strutture di Iterazione

Ripetere istruzioni: Mentre, Per, contatori e accumulatori.

1. Mentre (While) e Per (For)

Mentre ripete finché una condizione è vera (non sai quante volte). Per ripete un numero definito di volte con un contatore automatico.

Attenzione al ciclo infinito: se nulla, dentro il ciclo, avvicina la condizione alla fine, il programma non si ferma mai.

2. Contatori e accumulatori

Concetto	Esempio
Contatore	contatore = contatore + 1
Accumulatore	totale = totale + valore

Esercizio 1: Quante flessioni riesci a fare?

Ciclo Mentre risposta<>"basta": aumenta contatore, chiedi se continuare. Mostra il totale.

Esercizio 2: Tabellina Personalizzata

Chiedi numero. Per i=1 a 10: mostra numero & " x " & i & " = " & (numero*i).

Esercizio 3: Quanto spendi in snack?

totale=0. Per i=1 a 7: chiedi speseGiorno, totale=totale+speseGiorno. Mostra il totale della settimana.

Modulo 07 — Progetto Finale: Indovina il Numero

Variabili, sequenza, selezione e iterazione in un solo algoritmo.

Il computer sceglie un numero segreto tra 1 e 50. L'utente prova a indovinarlo, ricevendo indizi "troppo alto/troppo basso", finché non ci riesce. Alla fine si mostrano i tentativi impiegati.

1. Variabili e numero casuale

```
Declare numeroSegreto, tentativo, contatore : Integer
numeroSegreto = Random(50) + 1
contatore = 0
tentativo = 0
```

Random(50) restituisce un intero da 0 a 49: aggiungendo 1 si ottiene l'intervallo 1-50.

2. Il ciclo e gli indizi

```
Mentre tentativo <> numeroSegreto
  Input tentativo
  contatore = contatore + 1
  Se tentativo > numeroSegreto
    Output "Troppo alto!"
  Altrimenti Se tentativo < numeroSegreto
    Output "Troppo basso!"
```

3. Il risultato finale

```
Output "Hai indovinato in " & contatore & " tentativi!"
```

Sfide extra: aggiungi un numero massimo di tentativi, oppure mostra di quanto ci si è avvicinati invece di solo Alto/Basso.

CONSEGNA: gioca almeno 3 partite davanti all'insegnante, mostrando che il conteggio dei tentativi è corretto.