**2° lezione di programmazione 1**

**Diagrammi di flusso**

Prima va fatto il diagramma di flusso quindi si va a descrivere l’algoritmo e poi si procede alla sua codifica.

Il diagramma di flusso è formato:

* **Da blocchi** che rappresentano le azioni.
* **Da connettori** che specificano in quale ordine vanno eseguite le azioni.

Immagine che contiene schermata, cerchio, diagramma, Carattere

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, diagramma

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, linea

Descrizione generata automaticamente

La variabile bisogna immaginarsela come un cassetto (che poi scopriremo essere nel linguaggio C di vario tipo o come int che indica un numero intero nel momento in cui si dichiara questa variabile si occupa uno spazio in memoria RAM) che contiene il risultato di un espressione.

È formata da:

* **Nome o identificatore**
* **Valore 1,2,”pippo”,1.5 ecc**
* **Espressione:** combinazione di operatori aritmetici costanti (variabile che non cambia nel corso del esecuzione del programma) che restituisco un risultato che viene memorizzato nella variabile**. (Variabile <-Espressione).** La variabile è dimensionata a seconda di cosa si deve conservare

Oltre gli operatori aritmetici ci sono anche quelli di incremento e decremento che si utilizzano per aumentare o diminuire un variabile. (Es i 🡨i+1) sto incrementando il contenuto di uno (N.B. nei linguaggi di programmazione (come C) si utilizza una scrittura contratta i++ i— i+=2).

Si utilizzano anche le variabili che fungono da “accumulatori” che si utilizzano sommare moltiplicare ecc dei numeri dentro un ciclo (**while e do-while).**  Questo accumulatore prima va inizializzato solitamente a 0 sarà: somma🡨0. Ha il compito di **conservare il valore precedente** a cui va sommato/sottratto/moltiplicato **il nuovo**.

**(ES somma 🡨 somma+numero)**

**Scambio del contenuto di due variabili**

**Immagine che contiene testo, diagramma, schermata, Carattere

Descrizione generata automaticamente**

Si inizia leggendo il contenuto delle due variabili e si va ad utilizzare una variabile ausiliaria che ha il valore di x, si utilizza perché quando si assegna il valore di Y a X la X perde il suo valore quindi bisogna conservarlo in una variabile.

**Istruzioni condizionali**

Un istruzione in cui si valuta il valore di un predicato e in funzione di questo valore di predicato si eseguono una sequenza di azioni.

La preposizione è una frase della quale si può affermare la veridicità. Se alla proposizione si aggiunge una variabile questa diventa un predicato. Operazione che permette di determinare se il predicato ́e vero o falso, sostituendo alle variabili i loro valori attuali.

I valori VERO e FALSO sono valori logici o booleani.

Operatori relazionali

|  |  |  |
| --- | --- | --- |
| **Operatore** | **Diagrammi di flusso** | **Linguaggio C** |
| Uguale | = | = |
| Diverso | ≠ | != |
| Maggiore | > | > |
| Maggiore uguale |  | >= |
| Minore | < | < |
| Minore uguale |  | <= |

**Nel rombo si inserisce il predicato**.

Immagine che contiene diagramma, linea, Carattere, design

Descrizione generata automaticamente

Si parla in questo caso di **condizione doppia** perché c’è sia il ramo VERO che il ramo FALSO, invece, si parla di **condizione semplice** quando si ha solo il ramo VERO

**Cicli o loop**

Si utilizza quando è necessario eseguire un certo numero di iterazioni per raggiungere un certo tipo di calcolo.

Si vuole ripetere una certa azione (ISTRUZIONI) finche non si verifica una certa condizione o predicato. Se diventa vera si esce dal ciclo altrimenti si continua ad iterare fin quando non diventa vera.

Immagine che contiene testo, schermata, Carattere, linea

Descrizione generata automaticamente

**Notazione Lineare Strutturata (NLS)**

I diagrammi di flusso se ci si presentano algoritmi più complessi possono essere soggetti ad errori e risultare poco leggibili, per tale motivo si adotta la pseudo codifica.

**Sequenza**   **Selezione**

Immagine che contiene testo, schermata, Carattere, linea

Descrizione generata automaticamenteImmagine che contiene testo, schermata, diagramma, Carattere

Descrizione generata automaticamente

**Ciclo**

Immagine che contiene testo, schermata, diagramma, linea

Descrizione generata automaticamente  
è importante sottolineare che NLS non è un linguaggio di programmazione quindi poi per codificarlo in un linguaggio di programmazione si deve rispettare la sintassi di questo. Un trucchetto per capire con certezza la condizione del while si deve fare quello che rende falso il ciclo.

**HAND TRACING**

**Immagine che contiene testo, schermata, numero, software

Descrizione generata automaticamente**

è da intertenersi come una sorta di debbugging .

**I vettori o Array**

L’array è una struttura dati omogenea cioè sono presenti dati tutti dello stesso tipo. Gli elementi sono tutti della stessa dimensione e sono dello stesso tipo. ogni elemento è indicizzato. V[0] in questo modo si sta identificando l’elemento alla posizione 0 (gli indici partono da 0. Piu in generale V[i] indica l’elemento alla posizione i-esimo (i arriva ad n-1 cioè la dimensione del vettore meno uno perché si parte da 0). Supponendo di avere un array di dimensioni N=4 quindi da 0 a 3 e si vuole assegnare 10 all’array in posizione 2 sarà: V[2]🡨10. **Per scorrere gli array si utilizzano i cicli.** La condizione inserita nel while per scorrere l’array deve essere i<N dove N è la dimensione del vettore. Perché non si utilizza minore o uguale? Se fosse in questo modo arriverebbe alla posizione N che **non esiste**  perché si arriva ad N-1.