**9° lezione prog1**

**Hand Tracing**

Si utilizza per simulare manualmente l’esecuzione di programma. Si vanno a simulare i vari passi degli algoritmi e a comprendere gli eventuali errori semantici(logica).Per l’hand tracing si utilizza una tabella costituita da:

-   La prima riga con le variabili di interesse

-   Nelle successive righe si utilizzano per tenere traccia di queste

-   una nuova riga va creata ogni qual volta il valore di una variabile di interesse subisce una variazione (si utilizza una riga per ogni istruzione)

-   si riporta anche l’output a parte in una nuova riga

**Operatori logici**

Con gli operatori logici si possono combinare differenti predicati affinché producono un valore Boaleano true  o false.

* **L'operatore &&( understand understand) AND logico,** restituisce True se le due  condizioni sono vere. È un operatore binario perché ha due variabili.
* **L’operatore OR logico (||)** è binario, produce true se una sola delle variabili e’ vera
* **L'operatore  NOT (!)** è unario(si applica ad una sola variabile)

**AND Logico**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Risultato** |
| **FALSE** | **FALSE** | **FALSE** |
| **FALSE** | **TRUE** | **FALSE** |
| **TRUE** | **FALSE** | **FALSE** |
| **TRUE** | **TRUE** | **TRUE** |

**OR Logico**

|  |  |  |
| --- | --- | --- |
| **A** | **B** | **Risultato** |
| **FALSE** | **FALSE** | **FALSE** |
| **FALSE** | **TRUE** | **TRUE** |
| **TRUE** | **FALSE** | **TRUE** |
| **TRUE** | **TRUE** | **TRUE** |

**NOT**

|  |  |
| --- | --- |
| **A** | **RISULTATO** |
| **FALSE** | **TRUE** |
| **TRUE** | **FALSE** |

**Il costrutto WHILE()**

È composto dalla seguente sintassi:

* **parola chiave while**
* **() condizione o predicato**
* **{} tra parentesi graffe le istruzioni da eseguire**

while(condizione o predicato){  istruzione1;  }

La condizione viene valutata fin quando è uguale a TRUE quando è  FALSE esce dal ciclo. Inoltre la condizione viene valutata prima di ogni iterazione.Immagine che contiene testo, Carattere, schermata, bianco

Descrizione generata automaticamente

In questo pezzo di codice si avranno anno iterazioni, pero anno partirà da 1 .

**Il costrutto for()**

si usa quando si ha la certezza del numero di iterazioni che verranno eseguite.

**for(inizializzazione; condizione; aggiornamento)**

**{**

**Istruzioni;**

**}**

* **l’inizializzazione** viene eseguita solo una volta prima dell’ inizio del ciclo
* **la condizione o predicato** restituisce un valore  TRUE o FALSE come in tutti gli altri cicli(while() do-while()). Viene valutata prima di ogni iterazione
* **aggiornamento** si utilizza per incrementare o decrementare il valore più variabile o  di una

**for(i=0; i<0; i++)**

**{**

**Istruzioni;**

**}**

**Il ciclo a condizione finale do while**

Il ciclo do while prima esegue il contenuto del ciclo do{} e alla fine verifica le condizione di ripetizione,

* se la condizione è verificata il ciclo viene ripetuto
* se la condizione non è verificata il ciclo termina

**do**

**{**

**istruzioni;**

**}while (condizione di ripetizione)**

**Viene utilizzato quando ad esempio si deve verificare se l’input inserito dall’utente è corretto, perchè prima si procede con l'inserimento dei valori e poi si va a controllare.**

**Come si contano le iterazioni in un ciclo?**

Supponendo di avere due numeri A e B con A<B. Un ciclo avrà B-A interazioni.Immagine che contiene testo, Carattere, bianco, schermata

Descrizione generata automaticamente

Se invece nella condizione se avesse <=  Allora si avranno B- A+1 interazioni.

**L'istruzione break;**

 si utilizza all'interno del costrutto switch ma anche nei cicli while e do-while per interrompere l'esecuzione del ciclo.

**L'istruzione continue;**

 se inserita in un ciclo for forza l'inizio di una nuova  senza che siano eseguite le istruzioni seguenti.