

12BHD INFORMATICA – a.a. 2018/19

Esercitazione di Laboratorio 1

Obiettivi dell'esercitazione

- Disegnare flow-chart
- Prendere confidenza con l'ambiente di sviluppo, compilazione e debug.
- Eseguire programmi scritti in linguaggio C per acquisire da tastiera, manipolare e visualizzare a video valori numerici interi

Contenuti tecnici

- Definizione della funzione *main* in un programma C
 - Definizione di variabili intere (*int*) e loro utilizzo
 - Uso di strutture elementari nei flow chart.
-

Da risolvere preferibilmente in laboratorio

Esercizio 1. Utilizzando l'ambiente di sviluppo, scrivere, compilare ed eseguire il seguente programma in linguaggio C, verificando che non ci siano *errori* né *warning* in fase di compilazione

```
#include <stdio.h>
int main(void)
{
    int x , y, z;

    printf("Introduci un numero intero: ");
    scanf("%d", &x);
    y = 3;
    z = x/y;

    printf("%d/%d=%d\n", x, y, z);
    return 0 ;
}
```

Dopo averlo eseguito, esercitarsi con l'esecuzione passo a passo osservando il valore delle variabili x, y e z tramite i 'watch', provare con diversi valori: 0,9,15,20.

Esercizio 2. Disegnare il flow-chart per il calcolo del modulo (valore assoluto) di un numero; in particolare il programma dovrà:

- a) Acquisire da input un valore intero, positivo o negativo, e memorizzarlo in una variabile opportunamente definita.
- b) Stabilire utilizzando la struttura elementare *if-then(-else)* se tale variabile contiene un valore negativo e, in questo caso, trasformarlo nel corrispondente valore positivo
- c) Inviare in output il valore finale, ovvero il modulo del valore acquisito

Esercizio 3. Utilizzando l'ambiente di sviluppo, scrivere, compilare ed eseguire il seguente programma in linguaggio C, verificando che non ci siano *errori* né *warning* in fase di compilazione. Si disegni infine il relativo diagramma di flusso.

```
#include <stdio.h>

int main (void)
{
    int x;

    printf("Inserisci un numero: ");
    scanf("%d", &x);

    if (x>0)
        printf("Il valore %d e' positivo\n", x);
    else
        printf("Il valore %d e' negativo o pari a 0\n", x);

    return 0;
}
```

Dopo averlo eseguito, esercitarsi con l'esecuzione passo a passo osservando il valore della variabile x tramite 'watch', provare con diversi valori: 10, -10, 0, 9, - 15.

Da risolvere a casa

Esercizio 4. Disegnare il flow-chart per stabilire se un numero è primo; in particolare, la soluzione dovrà:

- Acquisire da input un valore intero da tastiera
- Utilizzare le strutture elementari *while-do* o *do-while* per stabilire tramite un ciclo se il numero è primo o meno
- A seconda dei casi, inviare in output un opportuno messaggio.

Nota1) Sforzarsi di realizzare un algoritmo "ben strutturato", senza salti: in questo caso l'utilizzo di una variabile logica (in gergo "flag", bandierina) può aiutare.

Nota2) Adottare sempre il metodo dei raffinamenti successivi: ad esempio, nella stesura iniziale considero come divisori tutti i numeri che precedono il numero dato. Successivamente analizzo nel dettaglio il problema, per vedere se posso limitare il numero di cicli.

Esercizio 5. Scrivere un programma che definisca 3 variabili intere chiamate *operand1*, *operand2* e *result*, e:

- Acquisisca da tastiera il valore di *operand1* e *operand2* tramite la funzione *scanf*
- Ne calcoli la somma e la salvi nella variabile *result*
- Visualizzi a video il valore della variabile *result* utilizzando la funzione *printf*

Consiglio: prima di dare il run, eseguire sempre il programma passo-passo, controllando le variabili con il watch. In particolare, verificare sempre che i valori introdotti da tastiera e i valori letti nelle variabili coincidano, e così per i valori in output (test di correttezza per gli specificatori di formato e altro...). Inoltre verificare se ci sono casi in cui l'operazione di somma sia errata (overflow).

- Esercizio 6. Scrivere un programma C in grado di risolvere un'equazione di primo grado espressa nella forma $ax+b=0$; in particolare il programma dovrà:
- a) Definire due variabili intere (*int*), *a* e *b* per memorizzare i coefficienti dell'equazione
 - b) Definire una variabile intera chiamata *x* in cui memorizzare il risultato dell'equazione
 - c) Acquisire da tastiera il valore dei coefficienti *a* e *b*
 - d) Calcolare il valore di *x* e visualizzarlo a video
- Approfondimento: cosa succede nel caso in cui il valore di *a* è uguale a 0?

- Esercizio 7. ¹ Realizzare un programma per la conversione da gradi Celsius a gradi Fahrenheit e viceversa. Il programma deve permettere di inserire una lettera 'C' o 'F' che indica la scala (Celsius o Fahrenheit) dei gradi; il valore dei gradi verrà inserito in seguito. Il programma deve calcolare e visualizzare i gradi Fahrenheit o Celsius rispettivamente. Se la relazione tra gradi Celsius e gradi Fahrenheit non è nota, cercarla in Internet.

¹ Questo esercizio sarà svolto in modo multimediale e inserito sul Portale, tra il materiale comune, nelle settimane successive.