

Corso di Elementi di Informatica

Espressioni semplici e condizionali

- gli esercizi proposti richiedono la conoscenza di alcune istruzioni di base del C che saranno introdotte nelle lezioni della settimana di lezione corrente.
- L'obiettivo principale dell'esercitazione è imparare ad utilizzare l'ambiente integrato (scrivere un sorgente, compilarlo, fare il linking e usare il debugger).
- Si consiglia di svolgere gli esercizi in due passi: specifica dell'algoritmo (a parole, ma in maniera non ambigua) e poi codificarlo in C (quando si possiedono già le conoscenze necessarie).

Numeri pari e dispari

1) Scrivere un programma che stabilisce se un numero intero è pari o dispari.

Prerequisiti:

L'operatore modulo: $x\%y$;
restituisce il resto della divisione di x per y

Analizziamo il problema usando il linguaggio naturale:

- Richiedi un numero
- Calcola il resto della divisione con 2

Numeri pari e dispari

Un'analisi più dettagliata

- Richiedi un numero
- Calcola il resto della divisione con 2
- Se il resto è $= 0$ stampa "E' pari"
- Altrimenti stampa "E' dispari"

Numeri pari e dispari

Linguaggio c:

```
#include <stdio.h>
main()
{
    int num, resto;
    printf("Immetti un numero intero: ");
    scanf("%d", &num);
    resto = num % 2;
    if (resto == 0)
        printf("Il numero e' pari");
    else
        printf("Il numero e' dispari");
}
```

Compatibilità di Tipo

- Editare il seguente programma in Linguaggio C e monitorare con lo strumento di debugger le variabili X, Y, Z, SUM e t.

Compatibilità di Tipo

```
#include <stdio.h>
main()
{
    int X,Y;
    unsigned int Z;
    float SUM;
    char t;

    X = 27; Y = 343; t = 'c';
    Z = X + Y - 300;
    X = Z / 10 + 23;
    Y = (X + Z) / 10 * 10;
    X = X + 70;
    Y = Y % 10;
    Z = Z + X - 70;
    SUM = Z * 10;
    t = t + X;
    Y = t;
}
```

Esercizio n.2

- Aprire l'ambiente integrato C.
- Creare un nuovo progetto **media** e all'interno di questo inserire un file **voto.c**
- Progettare un algoritmo che dati tre numeri (voti) *a*, *b*, *c* che possono assumere i valori interi da 18 a 30, calcoli la loro media.
- Codificare il programma in C e scriverlo in **voto.c**
- Procedere alle operazioni di Compile, Link e correggere eventuali errori.
- Seguire l'esecuzione del programma con l'uso del debugger (mettere sotto osservazione la variabile *media*).

Esercizio n.3

- Creare un nuovo progetto **media1** con un file **voto1.c**
- Progettare e Codificare in C (scrivendolo in **voto1.c**) un programma che permetta il calcolo della media di tre voti dati in trentesimi. La media deve essere fornita in centesimi o trentesimi a seconda del valore assunto da una variabile *scala* di tipo carattere.
- Seguire l'esecuzione con il debugger (mettere sotto osservazione la variabile *media*).
- Cosa succede se definite le variabili a cui sono assegnati i voti (*a,b,c*) di tipo int?
- E cosa invece se le definite di tipo *float*?
- In cosa differiscono e perché'?

Esercizio n.4

- Progettare e Codificare in C un programma che calcoli il valore assoluto di un numero *x*.

Esercizio n.5

- Progettare e Codificare in C il programma che data una quantità di soldi in lire restituisca la quantità corrispondente in euro (/2000) o in dollari (/2500), a seconda del valore assunto da una variabile *moneta* di tipo char.