

Esercizi II parte

Argomenti da sapere

- Sistemi Operativi
- Reti
- Assembler
- Pseudocodice
- C++

Assembler

- **Codici mnemonici** per le operazioni
 - LOAD, STORE, ADD, SUB
- **Nomi mnemonici (identificatori)** al posto degli indirizzi RAM per i dati
- Tipi dai dati: INT, FLOAT

Assembler: esempio

```
Z : INT ;  
X : INT 38 ;  
Y : INT 8 ;  
LOAD R0 X ;  
LOAD R1 Y ;  
ADD R0 R1 ;  
STORE R0 Z ;
```

dichiarazioni degli
identificatori dei dati

istruzioni assembler

Esempio: Legge due valori da tastiera, li somma e stampa il risultato della somma a video

```
X: INT;
Y: INT;
Ris: INT;
  READ INP X;
  READ INP Y;
  LOAD R0 X;
  LOAD R1 Y;
  ADD R0 R1;
  STORE R0 Ris;
  WRITE OUT Ris;
  STOP;
```

Provate a fare lo stesso esercizio in C++

Esercizio (iterazione)

- Individuare fase di inizializzazione, di modifica e la condizione di terminazione in questo ciclo e il corpo del ciclo. Quante volte viene eseguito il corpo del ciclo? Cosa viene stampato?

```
contatore ← 5;
while(contatore < 8) do
  (stampa contatore -1;
  contatore ← contatore +3; )
```

- Il corpo del ciclo viene eseguito una volta; viene stampato il valore 4

Esercizio (ricorsione)

```
procedure Test1(Contatore)
if (Contatore < 3)
then (stampa Contatore; Test1(Contatore+1))
```

```
procedure Test2(Contatore)
if (Contatore < 3)
then (Test2(Contatore+1); stampa Contatore)
```

- Ingresso 1: che uscita dalle due procedure?
Test1: 1,2
Test2: 2,1

Sol. esercizi (cicli for e while) -- 1

- Scrivere un ciclo while equivalente al ciclo for infinito

```
for(int i = 0; i > -2; i++)
  {cout<< i;}
```

```
Soluzione: int i=0;
           while(i>-2)
             { cout<<i; i++; }
```

- Trasformare il seguente ciclo while in un ciclo for equivalente:

```
int i = 1;
while (i < 20) {cout << i; i++}
```

```
Soluzione: for(int i=1; i<20; i++)
           { cout<<i;}
```

Sol. esercizi (cicli for e while) -- 2

- Cosa stampano in output i seguenti frammenti di programmi?
 - ```
int i = -7, k = 1;
while (i < -2) {i = i + k; cout << i << " "; k++;}
```

Stampa -6 -4 -1
  - ```
int i = -4, k = 3;
while (!(i == k + 1))
{i++; k--; cout << (k-i) << " "};
```

Stampa 5 3 1 -1
 - ```
for (int n = 0; n < 20; n = n + 2)
{ cout << n << " "};
```

Stampa 0 2 4 6 8 10 12 14 16 18

## Esercizi (if-then-else, if-then)

Cosa stampano i seguenti frammenti di codice?

- ```
int giorno=20, mese=12;
if ((giorno == 25) && (mese == 12))
    cout << "E' Natale!";
//non stampa niente
```
- ```
int giorno=20, mese=12;
if ((giorno == 25) || (mese == 12))
 cout << "ciao!";
//stampa ciao
```
- ```
int giorno=25, mese=12;
if ((giorno == 25) && (mese == 12))
    cout << "E' Natale!";
//stampa: E' Natale!
```