

Ripasso INFORMATICA

STRUTTURA DI PROGRAMMA

- commento univale // single rize
/* rize multiple */
- istruzioni di precompilazione #include <iostream>
- dichiarazione dello spazio da usare using namespace std
- dichiarazione della funzione principale void main (void)
- inizio blocco {
- dichiarazioni variabili int i;
i = 1;
- implementazione algoritmo cout << "uscisci intero",
cin >> i;
i++;
cout << i;
- fine blocco }

Dichiarazione di variabile

| Tipo | Memoria | Valori |
|---------------|---------------------|---------------------|
| char | 8 bit | -128 a 127, 0 a 255 |
| short int | 16 bit | -32768 a 32767 |
| int | dipende da computer | signed, unsigned |
| long int | 32 bit | |
| long long int | 64 bit | |
| float | 32 bit | |
| double | 64 bit | |
| bool | 1 bit | T/F |

float a
double a

Esempio

```
short int i, j // i e di tipo signed
i = 100;
```

```
short int k,
k = 40000, // errore
```

```
unsigned short int k,
k = 40000, // corretto
```

```
char a = 'A',
a = a + 3,
cout << a; // stampa 'D' a video
```

Gli alias

```
typedef ku-orient int
ku-orient velocita; // equivale a int velocita
```

```
using ku-orient = int;
ku-orient velocita;
```

Le costanti

```
const float PI_GRECO = 3.14,
#define PI_GRECO 3.14,
```

le casting

```
int alfa = 100,
float beta = alfa, // casting implicito
```

```
float alfa = 2.0,
int beta,
beta = (int) alfa, // casting esplicito
```

Esercizi

(alfa >= 5) && (alfa <= 10),

alfa = 4 ⇒ V

alfa = 11 ⇒ F

Restituire vero per 5 <= alfa <= 10
altrimenti restituire falso

```
int i,
i = 5 / 3, // i = 2
```

```
int i = 1,
0 < i, // i = 2
i == 0, // i = 0
```

```
int j = 3,
i != j, // i = i + j;
```

```
beta = ++delta + alfa--,
```

```
delta = delta + 1,
beta = delta + alfa,
alfa = alfa - 1,
```