

Fondamenti di Informatica e Laboratorio T-AB e Fondamenti di Informatica T1
Ingegneria Elettronica e Telecomunicazioni e
Ingegneria dell'Automazione
a.a. 2010/2011

Lab 04.1

Alcuni schemi...

Leggere interi positivi fino a inserimento di 0

Modo 1

```
int num1;

num1=-1;
scanf("%d", &num1);
while( num1!=0 ){
    if( (num1>0) {
        ... // logica
    }
    else{
        if( num1<0 ) {
            printf("Errore, numero negativo");
            printf("Inserisci num. positivo");
            scanf("%d", &num1);
        }
        else printf("Esco!"); //qui num1==0
    }
} // while
```

Modo 2

```
int num1;

num1=-1;
do {
    scanf("%d", &num1);
    while( num1<0 ) {
        printf("Errore, numero negativo");
        printf("Inserisci num. positivo");
        scanf("%d", &num1);
    }
    if( num1!=0 ){
        ... // logica
    }
} while( num1!=0 );
```

Lettura di singolo carattere su riga

Modo 1

(ipotesi: c'è SOLO
l'andare a capo)

```
char carattere;
```

```
scanf ("%c%c", &carattere);
```

```
/*Leggo 1 carattere e 1 separatore
```

- Il PRIMO, cioè il carattere immesso dall'utente, lo assegno a **carattere**
- Il SECONDO, l'"a capo" ('\n') lo consumo senza assegnarlo ad alcuna variabile

```
*/
```

Modo 2

(ipotesi: carattere seguito da altri
separatori/caratteri e poi "a capo")

```
char carattere;
```

```
scanf ("%c", &carattere);
```

```
while (getchar() != '\n');
```

```
/*Leggo N caratteri/separatori
```

- Il PRIMO, cioè il carattere immesso dall'utente, lo assegno a **carattere**
- Il SECONDO, l'"a capo" ('\n') lo consumo senza assegnarlo ad alcuna variabile

```
*/
```

Condizioni logiche

- la congiunzione (AND) di N proposizioni è vera solo se tutte le N proposizioni sono vere, falsa altrimenti
- la disgiunzione (OR) di N proposizioni è falsa se e solo se tutte le proposizioni sono false, vera altrimenti

Condizioni logiche: Leggi di De Morgan

Pur non avendo il tempo di dimostrarle, le 2 possono tornare utili le seguenti relazioni:

$$\text{!(condizione1 \&\& condizione2)} == \\ \text{!(condizione1) || !(condizione2)}$$

$$\text{!(condizione1 || condizione2)} == \\ \text{!(condizione1) \&\& !(condizione2)}$$