

Esercitazione di Reti degli elaboratori

- Prof.ssa Chiara Petrioli



SAPIENZA
UNIVERSITÀ DI ROMA

Corso di C

Christian Cardia, Gabriele Saturni

Lezione di ripasso...

- **Tipi di dato primitivi**
- **Gli Array**
- **Gli array multidimensionali**
- **Le stringhe**
- **Heap & Stack**
- **Funzioni**
- **I puntatori**
- **Allocazione dinamica**
- **Socket programming**



Spiegazione alla lavagna

Tipi di dato primitivi

- char
- int
- short
- long
- float
- double

Gli Array (1/2)

- Collezione di dati omogenei
- Memorizzati in celle successive di memoria

Dichiarazione

```
6 //1. array di int con 10 posizioni
7 int i [10];
8 //2. array di char di 5 posizioni
9 char c [5];
10 //3. array di int inizializzato
11 int numeri [ ] = {7,2,3,6};
12 //4. array di char inizializzato
13 char caratteri [ ] = {'h','e','l','l','o'};
14 //5. array di char inizializzato
15 char caratteri2 [ ] = "Hello";
```

Gli Array (2/2)

Accedere agli elementi di un Array

```
5 //dichiarazione array di int di 10 elementi
6 int array[10];
7 //imposto il primo elemento a 3
8 array[0] = 3;
9 //imposto il secondo elemento a -3
10 array[1] = -3;
11 //imposto il settimo elemento a 5
12 array[6] = 5;
13 //dichiaraz variabile int e gli assegno il primo elemento
14 int a = array[0];
15 //dichiaraz variabile int e gli assegno il valore 2
16 int b = 2;
17 //equivale a scrivere 'array[5] = 3'
18 array[a+b] = 3;
19 //equivale a scrivere 'array[6]=(array[6] + 2)'
20 array[a*b] += 2;
21 //stampo i valori
22 printf("array[5]=%d, array[6]=%d \n",array[5],array[6]);
```

- **Output:** *array[5] =3, array[6]=7*

Gli Array multidimensionali

Esempio

```
5     int matrice[2][3];
6
7     int riga = 0;
8     int colonna = 0;
9
10    for ( riga=0; riga<2; riga++ ) {
11        for ( colonna=0; colonna<3; colonna++ ) {
12            printf("Input [%d][%d] \n",riga,colonna);
13            scanf("%d",&matrice[riga][colonna]);
14        }
15    }
16
17    for ( riga=0; riga<2; riga++ ) {
18        for ( colonna=0; colonna<3; colonna++ ) {
19            printf("%d ",matrice[riga][colonna]);
20        }
21        printf("\n");
22    }
```

Le stringhe...

Array di char

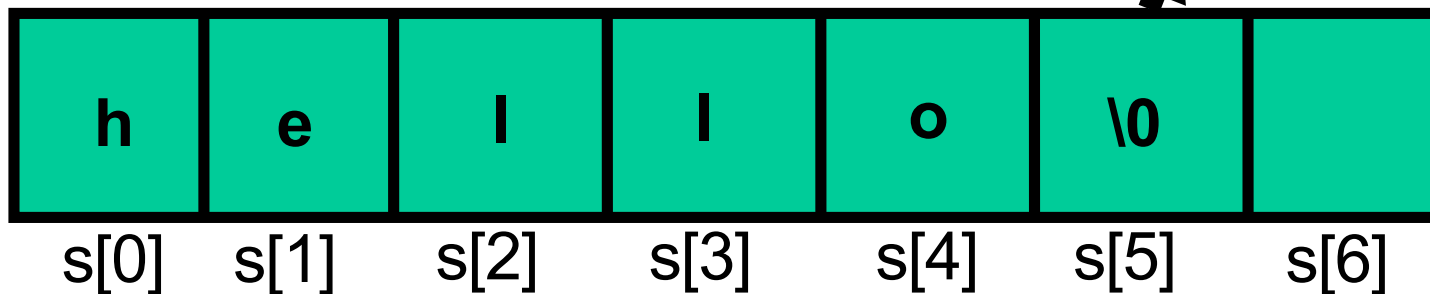
- Una stringa viene rappresentata come sequenza di caratteri che termina con '\0' (carattere terminatore)
- Quindi è un Array di *char* che termina con il carattere terminatore

- Dichiarazione:

```
char s[7] = "heLlo";
```

Cosa succede in memoria?

Il carattere "terminatore stringa" viene aggiunto automaticamente dal compilatore



Le stringhe...

Array di char

Step per leggere una stringa in input e stamparla:

- **Dichiarazione:**

```
char stringa[20];
```

Il vettore deve essere sufficientemente grande da contenere la stringa in input

- **Lettura in input:**

```
scanf("%s", stringa);
```

Legge i caratteri dall'input fino al primo spazio e li inserisce in *stringa*, aggiunge anche `\0`

- **Stampa in output:**

```
printf("%s \n", stringa);
```

Stampa ciò che è contenuto in *stringa* fino al carattere `\0` (escluso)

La libreria *string.h*

- Libreria di C che consente di manipolare le stringhe
- Per poter utilizzare le sue funzioni, nel nostro programma è opportuno importarla attraverso la direttiva:
- *#include <string.h>*
- Alcune delle funzioni che vedremo:

strlen

strcpy

strcmp

strcat

Heap & Stack

Funzioni

Puntatori

Allocazione Dinamica

- Spiegazione alla lavagna...

