

# Stringhe in C

Alessandra Giordani

[agiordani@disi.unitn.it](mailto:agiordani@disi.unitn.it)

Lunedì 20 maggio 2013

<http://disi.unitn.it/~agiordani/>



# EsErCiZiO PaRi DiSpArI

- Scrivere un programma che legge una frase e rende maiuscoli tutti i caratteri pari e minuscoli quelli dispari
- Per sapere se un numero è pari o dispari controlliamo il resto della divisione per 2
  - $3\%2 = 1$
  - $4\%2 = 0$
- Oppure sfruttiamo la divisione intera
  - $3/2*2 \neq 3$
  - $4/2*2 = 4$



# EsErCiZiO PaRi DiSpArl

- Possiamo procedere in più modi
  - Rendere tutto minuscolo e poi solo cambiare in maiuscolo i pari (o contrario)
  - Rendere tutti i dispari minuscoli e tutti i pari maiuscoli
    - 2 cicli separati (da  $i=0$  e da  $i=1$  con  $i=i+2$ )
    - 1 ciclo che considera coppie  $i$  e  $i+1$  a due a due



# Soluzione 1

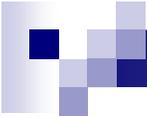
```
#include <stdio.h>
#define MAX 20
int main()
{
    char a[MAX];
    int i, dim;
    printf("Inserire una parola: ");
    scanf("%s", a);
    for(i=0;a[i]!='\0';i++)
        a[i]=toupper(a[i]);
    dim=i;
    for(i=1;i<dim;i=i+2)
        a[i]=tolower(a[i]);
    printf("Ecco la parola cambiata '%s'\n",a);
    return 0;
}
```

# Soluzione 2

```
#include <stdio.h>
#define MAX 20
int main()
{
    char a[MAX];
    int i, dim;
    printf("Inserire una parola: ");
    scanf("%s", a);
    for (i=0;a[i]!='\0';i++) ; //attenzione a questo ;
    dim=i;
    for (i=0;i<dim;i=i+2)
        a[i]=toupper(a[i]);
    for (i=1;i<dim;i=i+2)
        a[i]=tolower(a[i]);
    printf("Ecco la parola cambiata '%s'\n",a);
    return 0;
}
```

# Soluzione 3

```
#include <stdio.h>
#define MAX 20
int main()
{
    char a[MAX];
    int i, dim;
    printf("Inserire una parola: ");
    scanf("%s", a);
    for(i=0;a[i]!='\0';i++)
        if (i%2==0)
            a[i]=toupper(a[i]);
        else
            a[i]=tolower(a[i]);
    printf("Ecco la parola cambiata '%s'\n",a);
    return 0;
}
```



# Parole palindrome

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

Es. ANNA,  
ONORARONO, ...

```
main()
{
    char parola[32], i=0, n;

    printf("Inserisci una parola (lunga al max 31 caratteri): ");
    scanf("%31s", parola);
    n = strlen(parola);
    while((i <= n/2) && (parola[i] == parola[n-1-i]))
        i++;
    if(i > n/2)
        printf("La parola %s e' palindroma.\n", parola);
    else
        printf("La parola %s non e' palindroma.\n", parola);
    exit(0);
}
```

# Frase palindrome (a meno di spazi)

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

main()
{
    char parola[52], app[52], i = 0, j = 0, n;
    printf("Inserisci una frase (lunga al max 51 caratteri): ");
    scanf("%51[A-Za-z ']", parola);
    while(parola[i]!='\0')
    {
        if ((parola[i]!=' ') && (parola[i]!='\'))
        {
            app[j]=parola[i];
            j++;
        }
        i++;
    }
    app[j]='\0';
    n = j;
    i = 0;
    while((i <= n/2) && (app[i] == app[n-1-i]))
        i++;
    if(i > n/2)
        printf("La frase %s e' palindroma.\n", parola);
    else
        printf("La frase %s non e' palindroma.\n", parola);
    exit(0);
}
```

Es. i topi non avevano  
nipoti, ai lati d'italia...

# Inversione di una stringa

- Leggo una parola *a* di lunghezza sconosciuta (dim max 20)
- Inserisco in *b* i caratteri di *a* in ordine inverso
- Stampo stringhe

```
#include <stdio.h>
#define MAX 20
int main()
{
    char a[MAX], b[MAX];
    int i, dim;
    /*leggo parola*/
    printf("Inserire una parola (dim.max %d): ", MAX-1);
    scanf("%s", a);
    /*stampo parola - e calcolo lunghezza*/
    printf("Ho letto: ");
    for(i=0; a[i]!='\0'; i++)
        printf("%c", a[i]);
    dim=i;
    printf("' lunga %d caratteri\n", dim);
    /*inverto caratteri*/
    for(i=0; i<dim; i++)
        b[i]=a[dim-1-i];
    b[dim]='\0';
    /*stampo parola invertita*/
    printf("Ecco la parola invertita '%s'\n", b);
    return 0;
}
```



# Esecizi:

- Inversione di una frase
- Implementare i programmi visti in queste slides separando il codice relativo alla manipolazione delle stringhe in una funzione a parte
  - `void cambia (char * str)`
  - `int palindroma (char * str)`
  - `void inverti (char * str)`



# Simulazione di prova pratica

- La concatenazione di due stringhe è palindroma?
- Implementare un programma che richiama la funzione:
  - `int palindrome (char *s1, char *s2)`
- Che ritorna 1 se la concatenazione di s1 con s2 è una stringa palindroma
  - con “ai lati” e “d’italia” ritorna 1
  - con “d’italia” e “ai lati” ritorna 0