

# Laboratorio n.5

## Files, puntatori, ricorsione

9 e 16 giugno 2006

# Files

Scrivere un programma che

- Riceve su riga di comando il nome di un file da analizzare
- Apre il file e lo legge carattere per carattere contando le vocali che incontra (minuscole o maiuscole) – *scrivere una funzione `int isVocal(char a)` che ritorna 1 se il carattere passato come parametro è una vocale maiuscola o minuscola, altrimenti ritorna 0*
- Chiude il file, stampa il numero di vocali trovate e termina

# Files

- Scrivere un programma che apre in scrittura un file il cui nome va specificato come parametro su riga di comando, poi inizia a leggere da tastiera delle stringhe (usare gets) e le scrive sul file, fino a quando non ne viene inserita una di lunghezza nulla (usare strlen), al che chiude il file e termina.

# Puntatori: trova errore

- Correggere il programma e stabilire il valore finale delle variabili

```
main ()  
{   int a, b, *c;  
    a=8; b=9;  
    c=&a;  
    *c=a+5;  
    b=c;  
}
```

# Funzioni e passaggio parametri

- Quanto varranno i parametri nella funzione chiamante dopo l'esecuzione della funzione?

```
void f1 (int a, int b)  
{ int c;  
  a = a + 4;  
  c = a + b;  
  b = b - 3; }
```

# Parametri: trova errore

- La funzione dovrebbe scambiare due valori (ma non lo fa)

```
void scambia(int *a, int *b)  
{  
    int *temp;  
    temp = a;  
    a = b;  
    b = temp;  
}
```

# Puntatori

- Dichiarare una struttura con due campi interi a e b
- **Nel main** dichiarare una variabile x di quel tipo e leggerne da tastiera i valori dei campi con due blocchi printf-scanf
- Scrivere una funzione confronta, chiamata dal main, che stampa “UGUALI” o “DIVERSI” a seconda di come sono i valori di quei due campi
- Scrivere una funzione scambia, chiamata in modo *opportuno* dal main, che fa in modo che i valori dei campi a e b della variabile x del main

# Puntatori

- Scrivere una funzione `int mystrlen1(char * a)` che conta e ritorna il numero di caratteri di una stringa, come fa `strlen`.
- La funzione deve essere implementata usando esclusivamente puntatori (non usare array []) e in modo iterativo
- Implementare poi `int mystrlen2(char * a)` che lavora con logica ricorsiva:
  - Se il primo carattere della stringa è il “tappo” (`\0`), allora la lunghezza è 0: ritorna 0.
  - Altrimenti, la lunghezza della stringa è pari a 1+la lunghezza della sottostringa dal secondo carattere in poi (lunghezza ottenuta richiamando ricorsivamente `mystrlen2` su tale sottostringa)
- Scrivere un `main` che legge da tastiera una stringa e poi ne determina la lunghezza con entrambe le funzioni. Verificare che il risultato sia lo stesso nei due casi.