

Aula V

Dip. di Matematica “G. Castelnuovo”

Univ. di Roma “La Sapienza”

Array e puntatori; cenni sul debugging

Igor Melatti

Slides disponibili (assieme ad altro materiale) in:

<http://www.dsi.uniroma1.it/~melatti/programmazione1.2007.2008.html>

- Scrivere una funzione per determinare se in un array di interi vi sono solo elementi dispari.
- Scrivere una funzione per determinare se una tavola è ordinata crescentemente
- Scrivere una funzione iterativo per calcolare la media e la varianza di un array di reali.
- Scrivere una funzione iterativo per calcolare la media e la varianza di un array di reali in modo che ogni elemento dell'array venga ispezionato solo una volta.
- Scrivere una algoritmo per decidere se in un array con un numero di elementi pari  $t[0, \dots, 2N-1]$  gli elementi di indice pari sono ordinati in ordine crescente e quelli di indice dispari in ordine decrescente

- Scrivere una procedura che modifichi un array sommando 1 a tutti i suoi elementi
- Scrivere una funzione per contare il numero di cambi di segno in un array il cui primo elemento è differente da 0. Un cambio di segno si verifica quando nell'array vi sono due interi di segno opposto contigui o separati solo da 0.

- Supponiamo che vi siate accorti che un programma non “gira”
- Come si fa a trovare l’errore?
- Esistono programmi che aiutano a trovare i *bug*
- Infatti si chiamano debugger
- Sotto Linux c’è il gdb e la sua interfaccia grafica ddd

- Per usare il gdb, occorre aggiungere un'opzione a tempo di compilazione:  
-ggdb
- Quindi, anziché compilare scrivendo ad esempio  
`gcc nomeprogramma.c -o nomeprogramma`  
si scrive  
`gcc nomeprogramma.c -o nomeprogramma -ggdb`
- Dopodiché, se si ha il ddd si scrive `ddd nomeprogramma &`
- Altrimenti, si scrive `gdb nomeprogramma`

- Per prima cosa, occorre mettere un breakpoint
  - almeno finché non acquisite un po' di esperienza
- Per farlo, con ddd basta cliccare sulla linea in questione col tasto destro del mouse e scegliere `Set Breakpoint`
- Con gdb, occorre scrivere `b n`, mettendo al posto di `n` il numero della riga dove volete fermare l'esecuzione del programma
- Dopodiché, si può andare avanti passo passo o scrivendo `n` (gdb) o cliccando su `Next` (ddd)

- Per entrare dentro una funzione, premete `Step` (ddd) o scrivete `s` (gdb)
- Per far continuare il programma fino al prossimo breakpoint o fino alla fine premete `Cont` (ddd) o scrivete `c` (gdb)
- Per vedere quanto vale una variabile `v`, scrivete `display v` (gdb) o cliccate col destro sulla variabile desiderata e scegliete l'opzione opportuna (ddd)
- Per avere informazioni su un comando `cmd`, scrivere `help cmd`; ad esempio: `help breakpoint`

- Altri comandi interessanti:
  - sia per gdb che per ddd (quest'ultimo ha il gdb comunque presente in basso)
  - `bt` mostra l'intera sequenza di chiamate a funzione fino a quel momento
  - `up` si sposta nella chiamata a funzione superiore
  - `down` si sposta nella chiamata a funzione inferiore



Provate a correggere il file `soluzioni_esame_errato.c` usando il `gdb`