

Prova scritta di Sistemi di Elaborazione Ingegneria delle Telecomunicazioni

Ing. G. Lettieri

31 Gennaio 2006

1. Un semplice termostato collegato ad una caldaia il riscaldamento casalingo possiede tre ingressi e una uscita:

- un ingresso T_1 , che vale 1 se la temperatura dell'ambiente supera i 20°C , e 0 altrimenti;
- un ingresso T_2 , che vale 1 se la temperatura dell'ambiente supera i 16°C , e 0 altrimenti;
- un ingresso h , proveniente da un orologio, che vale 1 se l'ora è compresa tra le 7:00 e le 22:00, e 0 altrimenti;
- una uscita c , collegata all'interruttore della caldaia. Se l'uscita vale 1, la caldaia si accende, altrimenti si spegne.

Durante il giorno il termostato cerca di mantenere al temperatura ambiente intorno ai 20°C e, durante la notte, intorno ai 16°C . Disegnare il diagramma degli stati di una macchina di Moore che rappresenti il funzionamento del termostato. È necessaria una macchina sequenziale per risolvere questo problema? motivare la risposta.

2. Supponiamo di avere un programma scritto in parte in Assembler e in parte in C++

```
.data                                jg c1
s:      .long 0                       jmp c2
.text                                c1:   incl s
m:      pushl %ebp                    decl %eax
        movl %esp, %ebp              jmp c0
        pushl %eax                   c2:   popl %eax
        movl 8(%ebp), %eax           leave
c0:     cmpl $0, %eax                ret
```

```
void f()
{
    int i, j;

    for (i = 1; i <= 4; i++)
        for (j = 1; j <= 4; j++)
            m(i + j);
}
```

- (a) Dire quale sarà il contenuto della locazione `s` alla fine del programma.
(b) Tradurre la funzione `f` in Assembler.

3. Supponiamo di avere un file `elenco.txt`, in cui ogni linea contiene un cognome e una città (separati da un carattere di tabulazione). Dire quale sarà l'effetto del seguente script per la shell `sh` di Unix:

```
cut -f2 elenco.txt | sort | uniq > /tmp/f
for v in `cat /tmp/f`
do
    grep "$v" elenco.txt | cut -f1 > "$v"
done
```

4. Supponiamo di avere una cache associativa a 4 vie, con linee di 8 byte e una capacità totale di 32 byte. Dire quali dei seguenti accessi al byte causano *miss*, partendo da una cache vuota, se il rimpiazzamento è di tipo FIFO:

57, 35, 5, 34, 30, 50, 60, 1, 24, 23.

Mostrare quindi lo stato della cache dopo l'ultimo accesso.