

Prova scritta di Sistemi di Elaborazione Ingegneria delle Telecomunicazioni

Ing. G. Lettieri

22 Luglio 2005

1. Progettare il grafo degli stati di una macchina di Mealy minima con 3 stati di ingresso (0, 1 e +) e quattro stati di uscita: 0, 1, 2, *E*. Se gli ultimi tre stati di ingresso sono del tipo $n + m$ (con n e m che possono valere 0 o 1), la macchina restituisce in uscita il valore della somma. In tutti gli altri casi, la macchina restituisce *E*.
2. Supponiamo di avere il seguente programma, scritto in parte in Assembler e in parte in C++:

```
.data                                cmpl $0, (%esi)
beg:  .long 15                        je c1
      .long 0                        cmpl $1, (%esi)
      .long 10                       je c2
      .long 1                        addl $8, %esp
      .long 20                       jmp fine
      .long 2                        c1:  call f1
end:  .long 40                       jmp c3
      .text                          c2:  call f2
main: movl $0, %eax                  c3:  addl $8, %esp
      movl $beg, %esi                addl $4, %esi
c0:  pushl %eax                      cmpl $end, %esi
      pushl (%esi)                   jle c0
      addl $4, %esi                  fine: ret
```

```
int f1(int a, int b)
{
    return a + b;
}
int f2(int a, int b)
{
    return a - b;
}
```

- (a) Dire quale sarà il contenuto del registro `%eax` quando l'esecuzione raggiunge l'istruzione con etichetta `fine`;
- (b) Tradurre le funzioni `f1` e `f2` in Assembler.

3. La directory corrente di un sistema Unix contiene il seguente sottoalbero:

```
dir
|-- aaa
|   |-- aba
|   |-- aca
|   '-- bbe
|-- bbb
|   '-- adc
'-- ccc
    |-- acb
    '-- acd
```

Consideriamo il seguente script per la shell sh:

```
N=0
for f in dir/*/*
do
    if echo "$f" | grep "$1" > /dev/null
    then
        echo "$f" | sed "s/$1/X/g"
    else
        N='expr $N + 1'
    fi
done
echo $N
```

Dire quale sarà l'output dello script, eseguito nella directory corrente, con parametro "b".

4. Supponiamo di avere una cache completamente associativa, con due linee di un byte ciascuna, e la seguente sequenza di accessi ai byte:

0, 1, 2, 3 (ripetuta 1000 volte)

- (a) Calcolare il numero di *miss* nel caso in cui la cache utilizzi l'algoritmo di rimpiazzamento LRU;
- (b) Calcolare il numero di *miss* nel caso in cui la cache utilizzi l'algoritmo di rimpiazzamento MRU (che rimpiazza sempre la pagina usata più recentemente).