

# Compito di Laboratorio di Calcolatori - Prof. G. Ascia

Cognome e Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

## Quesito n. 1: (21 punti)

Creare un file di testo utilizzando la convenzione **cognome.nome.s** (es. *rossi.paolo.s*).  
Scrivere un programma in linguaggio Assembly MIPS che traduce il seguente programma C:

```
#include <stdio.h>

main() {
    int A[5]={1,2,2,1,4}, B[5],C[5], i;

    for(i=0;i<5;i++)
        {printf("Inserisci un numero: ");
         scanf("%d",&B[i]);
        }

    for(i=0;i<5;i++)
        if(B[i]/ A[i] > 5 )
            C[i]= B[i]/2;
        else C[i]= B[i]+A[i];

    for(i=0;i<5;i++)
        printf("B[%d]= %d",i,B[i]);
    for(i=0;i<5;i++)
        printf("C[%d]= %d",i,C[i]);
}
```

## Quesito n. 2: (6 punti)

Con riferimento al programma della quesito n. 1, e considerando una frequenza della CPU  $f = 200$  Mhz, calcolare CPI e  $t_{CPU}$  (comprensivo di unità di misura) nei casi seguenti.

- cache dati e istruzioni ideale (senza miss)
- cache L1 unificata di 4 K, blocco 32 byte, associativita' 2 miss penalty 40 cicli;
- cache istruzioni di 2 KByte, blocco 32 byte, associativita' 4, miss penalty 40 cicli; cache dati di 2 Kbyte, blocco 16 byte, associativita' 4, miss penalty 30 cicli.

## Quesito n. 3: (5 punti)

Descrivere la soluzione della Victim Cache per ridurre il miss rate.

# Compito di Laboratorio di Calcolatori - Prof. G. Ascia

Cognome e Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

## Quesito n. 1: (21 punti)

Creare un file di testo utilizzando la convenzione **cognome.nome.s** (es. *rossi.paolo.s*).  
Scrivere un programma in linguaggio Assembly MIPS che traduce il seguente programma C:

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

main() {
    char S[6];
    int A[5], C[5], dim, i;

    do {
        printf("Inserisci una stringa con 5 cifre positive minori di 6");
        gets(S);
        dim=strlen(S);
        if(dim==5)
            for(i=0; i<dim; i++)
                if(S[i]< 49 || S[i]> 53)
                    break;
    } while (dim < 5 || i < dim);

    for(i=0; i<5; i++)
        {printf("Inserisci un numero: ");
        scanf("%d", &A[i]);
        C[i]=A[i]+S[i]-48;
        }

    for(i=0; i<5; i++)
        printf("C[%d]= %d", i, C[i]);
}
```

## Quesito n. 2: (6 punti)

Con riferimento al programma della quesito n. 1, e considerando una frequenza della CPU  $f = 250$  Mhz, calcolare CPI e  $t_{CPU}$  (comprensivo di unità di misura) nei casi seguenti.

- cache unica ideale (senza miss)
- cache L1 unificata di 8 K, blocco 32 byte, associativita' 1 miss penalty 40 cicli;
- cache istruzioni di 4 KByte, blocco 32 byte, associativita' 4, miss penalty 35 cicli; cache dati di 4 Kbyte, blocco 32 byte, associativita' 4, miss penalty 35 cicli.

## Quesito n. 3: (5 punti)

Descrivere la soluzione della Victim Cache per ridurre il miss rate.

# Compito di Laboratorio di Calcolatori - Prof. G. Ascia

Cognome e Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

## Quesito n. 1: (21 punti)

Creare un file di testo utilizzando la convenzione **cognome.nome.s** (es. *rossi.paolo.s*). Scrivere un programma in linguaggio Assembly MIPS che traduce il seguente programma C:

```
#include <stdio.h>

main() {
    int A[5]={8,16,4,18,2};
    int B[5],C[5], i;

    for(i=0;i<5;i++)
        {printf("Inserisci un numero: ");
         scanf("%d",&B[i]);
        }

    for(i=0;i<5;i++)
        if(B[i]> A[i] )
            C[i]= (B[i]-A[i])/2;
        else C[i]= B[i]/2;

    for(i=0;i<5;i++)
        printf("B[%d]= %d",i,B[i]);
    for(i=0;i<5;i++)
        printf("C[%d]= %d",i,C[i]);
}
```

## Quesito n. 2: (6 punti)

Con riferimento al programma della quesito n. 1, e considerando una frequenza della CPU  $f = 250$  Mhz, calcolare CPI e  $t_{CPU}$  (comprensivo di unità di misura) nei casi seguenti.

- memoria unica senza stalli (caso ideale);
- cache L1 unificata di 4 K, blocco 32 byte, associativita' 2 miss penalty 20 cicli;
- cache istruzioni di 2 KByte, blocco 32 byte, associativita' 4, miss penalty 50 cicli; cache dati di 2 KByte, blocco 16 byte, associativita' 4, miss penalty 30 cicli.

## Quesito n. 3: (5 punti)

Descrivere la soluzione della Victim Cache per ridurre il miss rate.

# Compito di Laboratorio di Calcolatori - Prof. G. Ascia

Cognome e Nome: \_\_\_\_\_ Matricola: \_\_\_\_\_

## Quesito n. 1: (21 punti)

Creare un file di testo utilizzando la convenzione **cognome.nome.s** (es. *rossi.paolo.s*). Scrivere un programma in linguaggio Assembly MIPS che traduce il seguente programma C:

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

main() {
    char S[10];
    int A[6],C[6], dim,i;

    do {
        printf("Inserisci una stringa con 6 cifre positive minori di 8");
        gets(S);
        dim=strlen(S);
    } while (strlen(S)!=6);

    for(i=0;i<dim;i++)
        if(S[i]< 49 || S[i]> 53)
            S[i] = 48;

    for(i=0;i<5;i++)
        {printf("Inserisci un numero: ");
        scanf("%d",&A[i]);
        C[i]=A[i]+S[i]-48;
        }

    for(i=0;i<5;i++)
        printf("C[%d]= %d",i,C[i]);
}
```

## Quesito n. 2: (6 punti)

Con riferimento al programma della quesito n. 1, e considerando una frequenza della CPU  $f = 100$  Mhz, calcolare CPI e  $t_{CPU}$  (comprensivo di unità di misura) nei casi seguenti.

- memoria senza stalli (caso ideale);
- cache L1 unificata di 64 K, blocco 8 byte, associativita' 4 miss penalty 30 cicli;
- cache istruzioni di 32 KByte, blocco 8 byte, associativita' 1, miss penalty 30 cicli; cache dati di 32 KByte, blocco 4 byte, associativita' 1, miss penalty 30 cicli.

## Quesito n. 3: (5 punti)

Descrivere la soluzione della Victim Cache per ridurre il miss rate.