



# ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΜ&ΜΥ

Προηγμένα Θέματα  
Αρχιτεκτονικής Υπολογιστών

4<sup>η</sup> Άσκηση  
Ακ. έτος 2011-2012

Γρηγόρης Λύρας Α.Μ.: 03109687

18 Ιουλίου 2012

# Εισαγωγή

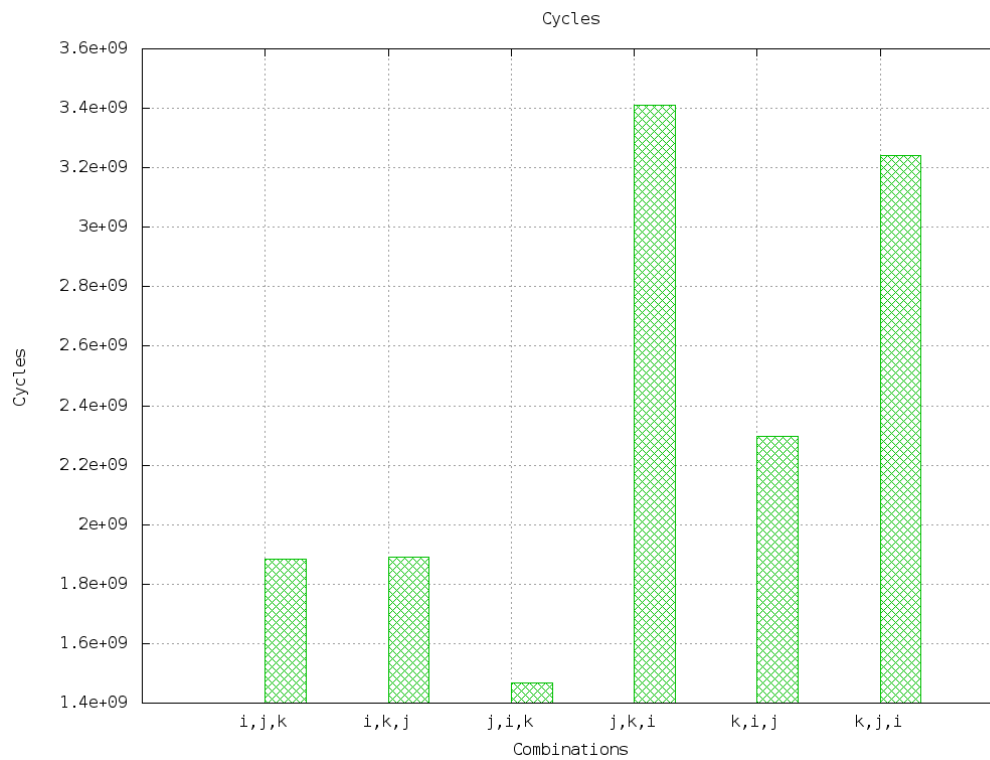
## I) Προσομοίωση

```
1  /* .....
2  * File Name : partA.c
3  * Creation Date : 16-07-2012
4  * Last Modified : Mon 16 Jul 2012 01:46:39 PM EEST
5  * Created By : Greg Liras <gregliras@gmail.com>
6  * .....*/
7
8  #include <stdio.h>
9  #include <stdlib.h>
10 #define __MAGIC_CASSERT(p) do { \
11     typedef int __check_magic_argument[(p) ? 1 : -1]; \
12 } while (0)
13
14 #define MAGIC(n) do { \
15     __MAGIC_CASSERT(!(n)); \
16     __asm__ __volatile__ ("xchg %bx,%bx"); \
17 } while (0)
18
19 #define MAGIC_BREAKPOINT MAGIC(0)
20
21
22
23 inline int min(int a, int b)
24 {
25     if(a<=b) return a;
26     else return b;
27 }
28 void init_matrix(float **mat, int n)
29 {
30     unsigned int i,j;
31     for(i=0; i<n; i++)
32         for(j=0; j<n; j++)
33             mat[i][j] = (float)(i+j);
34 }
35 int main(int argc, char **argv)
36 {
37     float **A,**B,**C;
38     int i,j,k,N;
39     N=atoi(argv[1]);
40     A=(float**)malloc(N*sizeof(float*));
41
42     for(i=0; i<N; i++)
43         A[i]=(float*)malloc(N*sizeof(float));
44
45     B=(float**)malloc(N*sizeof(float*));
46
47     for(i=0; i<N; i++)
48         B[i]=(float*)malloc(N*sizeof(float));
49
50     C=(float**)malloc(N*sizeof(float*));
51
52     for(i=0; i<N; i++)
53         C[i]=(float*)malloc(N*sizeof(float));
54
55     fprintf(stderr, "Initializing matrices...\n");
56     init_matrix(A, N);
57     init_matrix(B, N);
58     init_matrix(C, N);
59     MAGIC_BREAKPOINT;
60     for(i=0; i<N; i++) {
61         for(j=0; j<N; j++)
62             for(k=0; k<N; k++)
63                 C[i][j] += A[i][k]*B[k][j];
64     }
65     MAGIC_BREAKPOINT;
66     return 0;
67 }
```

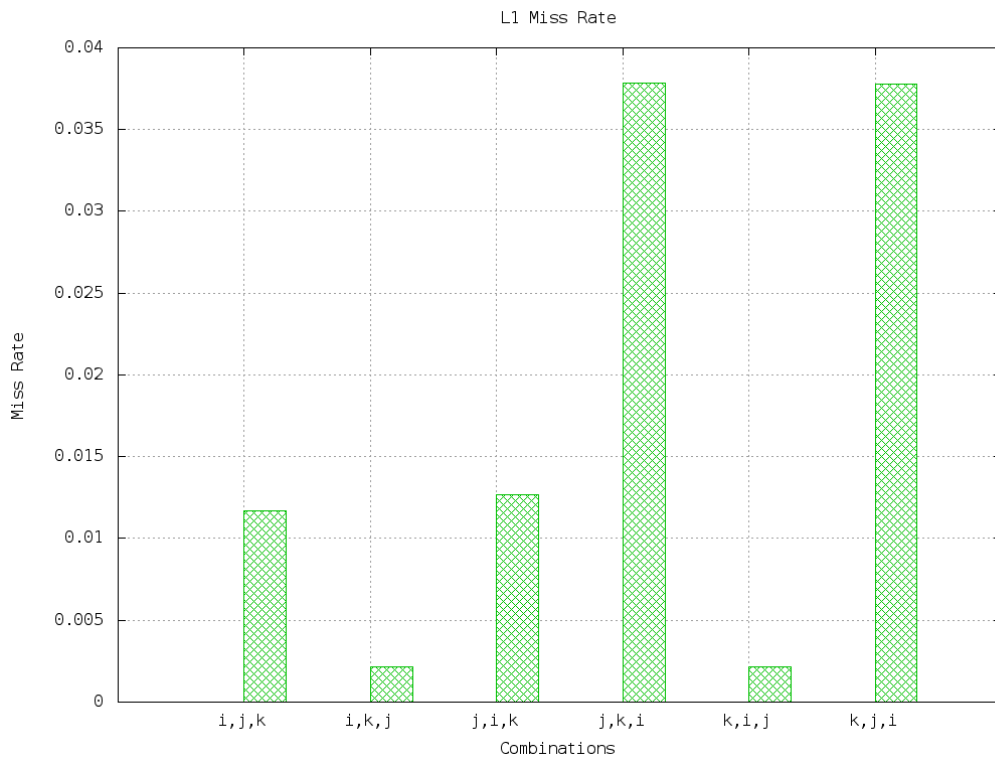
## II) Ιεραρχία μνήμης και μοντέλο απόδοσης

### Τεχνικές Βελτιστοποίησης

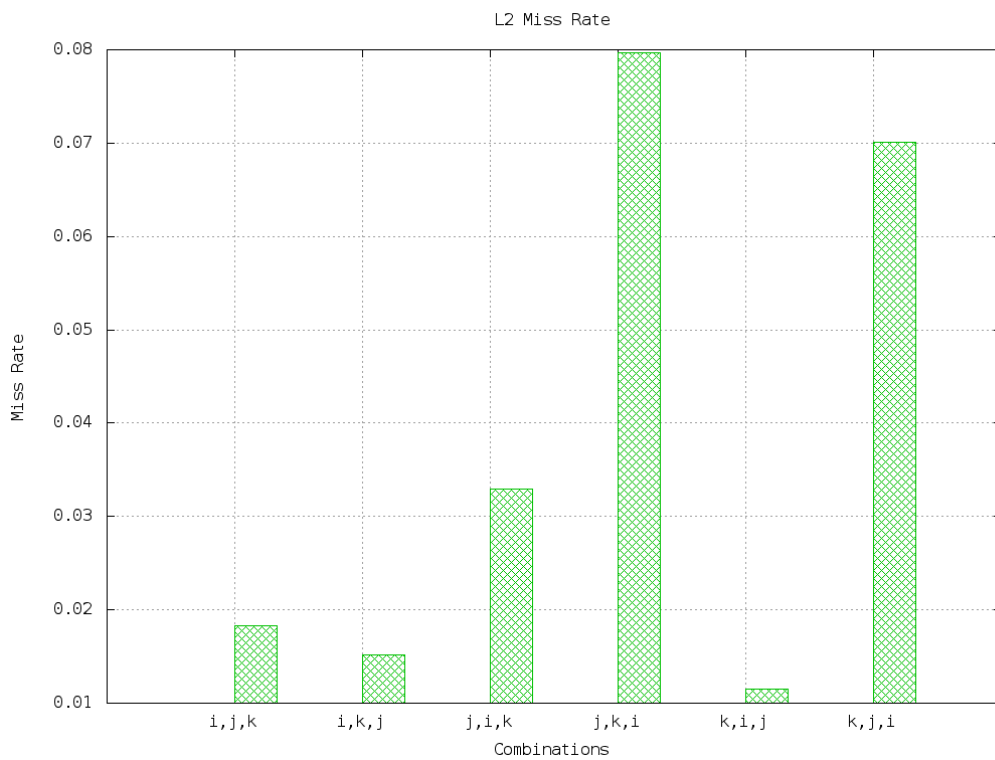
## III) Loop Interchange



Σχήμα 1: Cycles



Σχήμα 2: L1 Miss Rate



Σχήμα 3: L3 Miss Rate

## IV) Cache Blocking

```

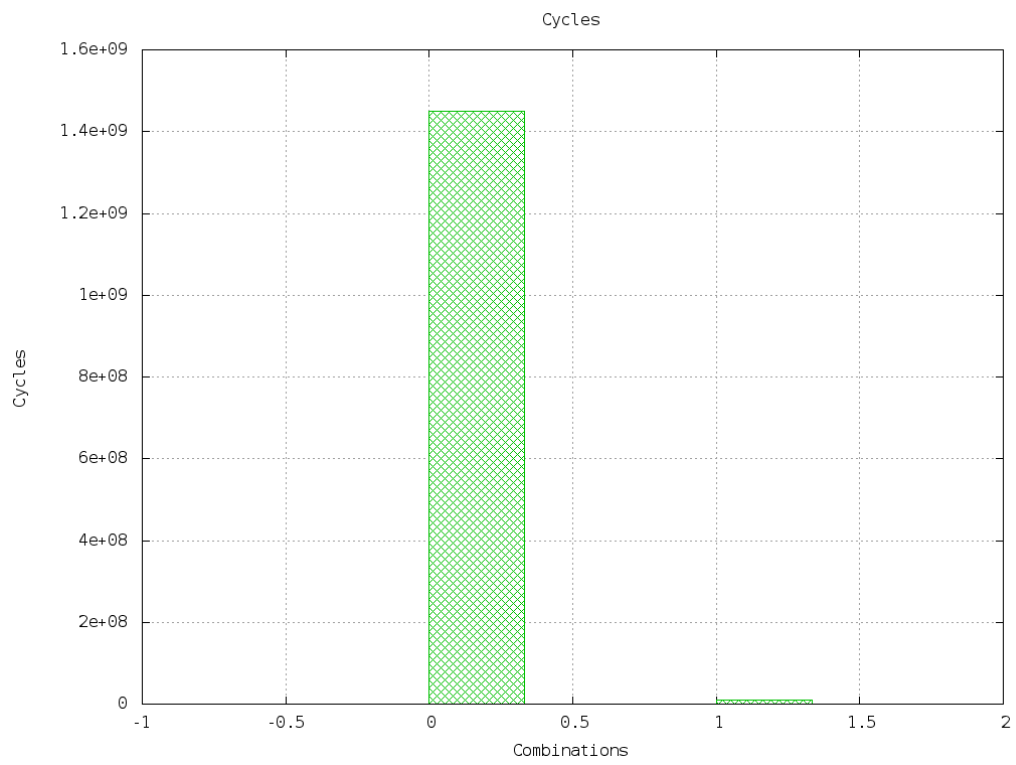
1  /* .....
2  * File Name : partA.c

```

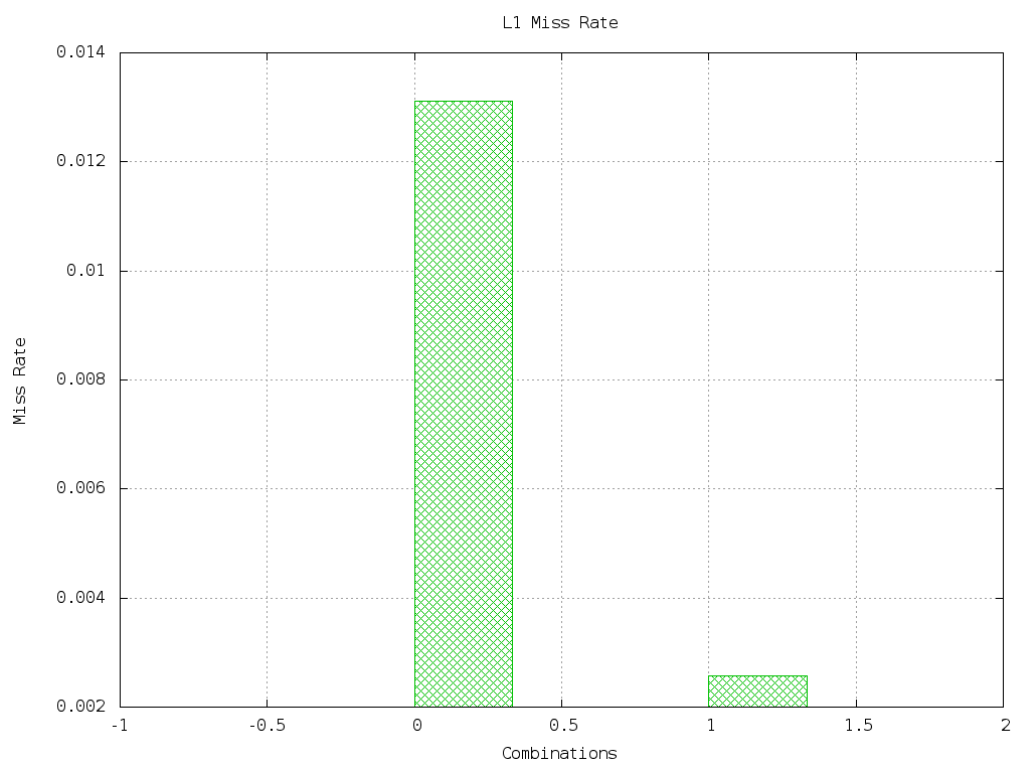
```

3  * Creation Date : 16-07-2012
4  * Last Modified : Tue 17 Jul 2012 06:07:55 PM EEST
5  * Created By : Greg Liras <gregliras@gmail.com>
6  .....*/
7
8  #include <stdio.h>
9  #include <stdlib.h>
10 #define __MAGIC_CASSERT(p) do { \
11     typedef int __check_magic_argument[(p) ? 1 : -1]; \
12 } while (0)
13
14 #define MAGIC(n) do { \
15     __MAGIC_CASSERT(!(n)); \
16     __asm__ __volatile__ ("xchg %bx,%bx"); \
17 } while (0)
18
19 #define MAGIC_BREAKPOINT MAGIC(0)
20
21
22
23 inline int min(int a, int b)
24 {
25     if(a<=b) return a;
26     else return b;
27 }
28 void init_matrix(float **mat, int n)
29 {
30     unsigned int i,j;
31     for(i=0; i<n; i++)
32         for(j=0; j<n; j++)
33             mat[i][j] = (float)(i+j);
34 }
35 int main(int argc, char **argv)
36 {
37     float **A,**B,**C;
38     int i,j,k,N;
39     int start,stop;
40     N=atoi(argv[1]);
41     A=(float**)malloc(N*sizeof(float*));
42
43     for(i=0; i<N; i++)
44         A[i]=(float*)malloc(N*sizeof(float));
45
46     B=(float**)malloc(N*sizeof(float*));
47
48     for(i=0; i<N; i++)
49         B[i]=(float*)malloc(N*sizeof(float));
50
51     C=(float**)malloc(N*sizeof(float*));
52
53     for(i=0; i<N; i++)
54         C[i]=(float*)malloc(N*sizeof(float));
55
56     fprintf(stderr, "Initializing matrices...\n");
57     init_matrix(A, N);
58     init_matrix(B, N);
59     init_matrix(C, N);
60     MAGIC_BREAKPOINT;
61     for(start=0; start<N; start+=16) {
62         stop = start + 16;
63         stop = stop <= N ? stop : N;
64         for(j=start; j<stop; j++)
65             for(i=start; i<stop; i++)
66                 for(k=start; k<stop; k++)
67                     C[i][j] += A[i][k]*B[k][j];
68     }
69     MAGIC_BREAKPOINT;
70     return 0;
71 }

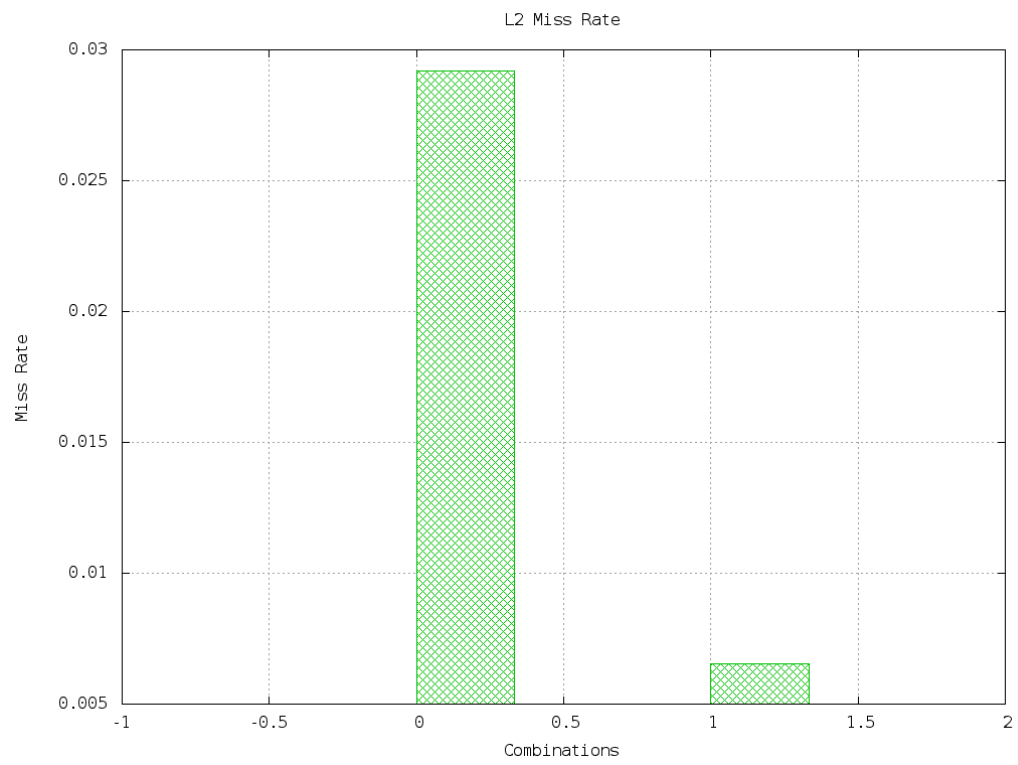
```



Σχήμα 4: Cycles



Σχήμα 5: L1 Miss Rate



Σχήμα 6: L3 Miss Rate