

# BÁO CÁO TUẦN 14

Bài 1 week 12

Code

E:\BTKTMT\week14\B1\_w12 - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

Ba1w12.asm B1\_w12

```

1  #Laboratory Exercise 7, Home Assignment 4
2  .data
3  Message: .asciiz "Ket qua tinh giai thua la: "
4  .text
5  main: jal WARP
6
7  print: add $a1, $v0, $zero # $a0 = result from N!
8  li $v0, 56
9  la $a0, Message
10 syscall
11 quit: li $v0, 10 #terminate
12 syscall
13 endmain:
14 #-----
15 #Procedure WARP: assign value and call FACT
16 #-----
17 WARP: sw $fp, -4($sp) #save frame pointer (1)
18 addi $fp, $sp, 0 #new frame pointer point to the top (2)
19 addi $sp, $sp, -8 #adjust stack pointer (3)
20 sw $ra, 0($sp) #save return address (4)
21
22 li $a0, 6 #load test input N
23 jal FACT #call fact procedure
24 nop
25
26 lw $ra, 0($sp) #restore return address (5)
27 addi $sp, $fp, 0 #return stack pointer (6)
28 lw $fp, -4($sp) #return frame pointer (7)
29 jr $ra
30 wrap_end:
31 #-----
32 #Procedure FACT: compute N!
33 #param[in] $a0 integer N
34 #return $v0 the largest value
35 #-----
36 FACT: sw $fp, -4($sp) #save frame pointer
37 addi $fp, $sp, 0 #new frame pointer point to stack's top
38 addi $sp, $sp, -12 #allocate space for $fp, $ra, $a0 in stack
39 sw $ra, 4($sp) #save return address
40 sw $a0, 0($sp) #save $a0 register
41
42 slti $t0, $a0, 2 #if input argument N < 2
43 beq $t0, $zero, recursive #if it is false ((a0 = N) >= 2)
44 nop
45 li $v0, 1 #return the result N!=1
46 j done
47 nop
48 recursive:
49 addi $a0, $a0, -1 #adjust input argument
50 jal FACT #recursive call
51 nop
52 lw $v1, 0($sp) #load a0
53 mult $v1, $v0 #compute the result
54 mflo $v0
55 done: lw $ra, 4($sp) #restore return address
56 lw $a0, 0($sp) #restore a0
57 addi $sp, $fp, 0 #restore stack pointer
58 lw $fp, -4($sp) #restore frame pointer
59 jr $ra #jump to calling
60 fact_end:
61

```

Line: 1 Column: 1 Show Line Numbers

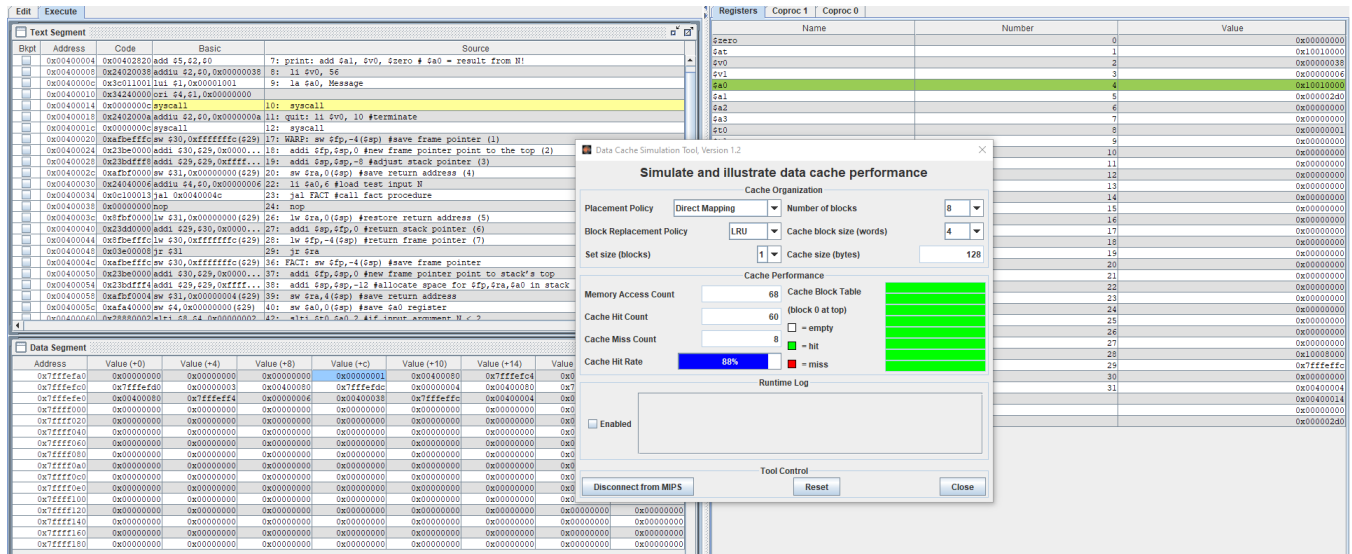
Mars Messages Run I/O

Coproc 0

...	Val...
.. 0 ..	...
.. 1 ..	...
.. 2 ..	...
.. 3 ..	...
.. 4 ..	...
.. 5 ..	...
.. 6 ..	...
.. 7 ..	...
.. 8 ..	...
.. 9 ..	...
.. 10 ..	...
.. 11 ..	...
.. 12 ..	...
.. 13 ..	...
.. 14 ..	...
.. 15 ..	...
.. 16 ..	...
.. 17 ..	...
.. 18 ..	...
.. 19 ..	...
.. 20 ..	...
.. 21 ..	...
.. 22 ..	...
.. 23 ..	...
.. 24 ..	...
.. 25 ..	...
.. 26 ..	...
.. 27 ..	...
.. 28 ..	...
.. 29 ..	...
.. 30 ..	...
.. 31 ..	...
pc	...
hi	...
lo	...

9:17 SA  
10/07/2022

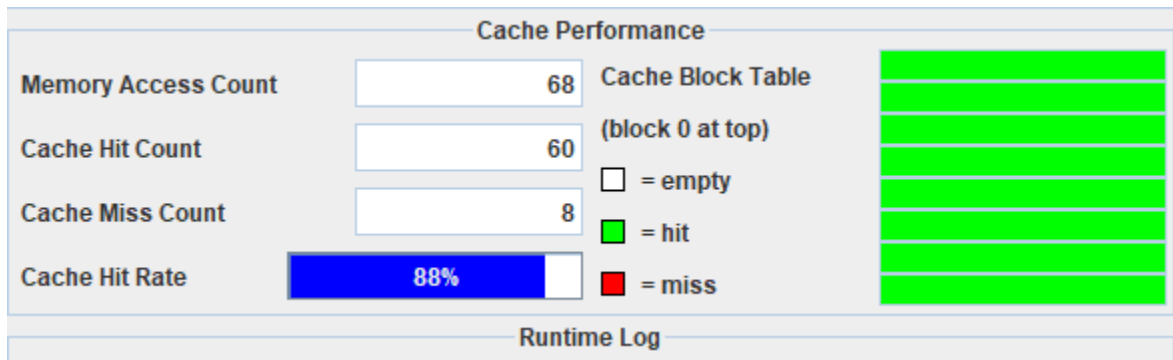
## Chạy trên Data Cache Simulator



## Phần 1 : giải thích dữ liệu bảng

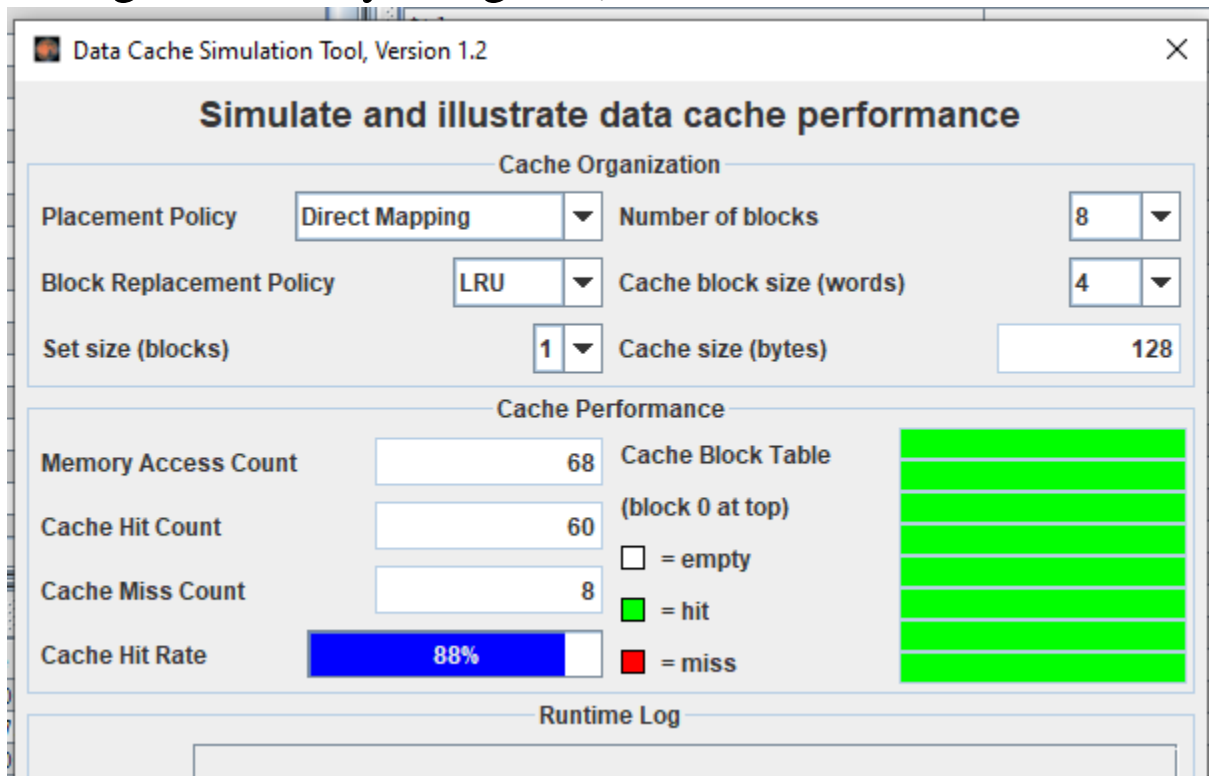
- Cache size (bytes): kích thước hay dung lượng của Cache Memory (= Number of blocks \* Cache block size)
- Cache block size (Words) : kích thước hay dung lượng của 1 block, như trong Mips mặc định words là 4 bytes
- Number of Blocks: số lượng block cho Cache Memory
- Cache Hit Rate: tỷ lệ truy cập thành công vào Cache Memory ( = Hit/Memory Access Count)
- Memory Access Count: số lần CPU yêu cầu truy cập vào Cache Memory (Tổng của Hit và Miss)
- Cache Hit count: số lần CPU yêu cầu truy cập vào Cache Memory thành công - Cache Miss count: số lần CPU yêu cầu truy cập vào Cache Memory thất bại và phải truy cập vào bộ nhớ chính (Ram).

- Biểu diễn và mô tả hoạt động của Cache được hiển thị tại Cache Block table:



Khi Number of block tăng lên thì dẫn tới kích thước của Cache tăng lên -> Cache Hit count tăng lên và Miss count giảm xuống.

- Ví dụ dưới đây thể hiện sự thay đổi khi chạy cùng 1 mã nguồn và thay đổi giá trị khi Number of block = 8



Ta thay Number of blocks bằng 64 ta được bảng sau:

**Data Cache Simulation Tool, Version 1.2**

**Simulate and illustrate data cache performance**

**Cache Organization**

Placement Policy: Direct Mapping | Number of blocks: 64

Block Replacement Policy: LRU | Cache block size (words): 4

Set size (blocks): 1 | Cache size (bytes): 1024

**Cache Performance**

Memory Access Count: 73

Cache Hit Count: 65

Cache Miss Count: 8

Cache Hit Rate: 89%

Cache Block Table (block 0 at top)

- ☐ = empty
- ☒ = hit
- ☐ = miss

**Runtime Log**

☐ Enabled

**Tool Control**

Disconnect from MIPS | Reset | Close

Cache hit count tăng lên

Memory Access Count tăng lên và Cache hit rate tăng lên

Vậy khi chuyển Number of block = 4

**Data Cache Simulation Tool, Version 1.2**

**Simulate and illustrate data cache performance**

**Cache Organization**

Placement Policy: Direct Mapping | Number of blocks: 4

Block Replacement Policy: LRU | Cache block size (words): 4

Set size (blocks): 1 | Cache size (bytes): 64

**Cache Performance**

Memory Access Count: 73 | Cache Block Table (block 0 at top): 4 green blocks

Cache Hit Count: 63 | ☐ = empty, ☒ = hit, ☐ = miss

Cache Miss Count: 10

Cache Hit Rate: 86%

**Runtime Log**

☐ Enabled

**Tool Control**

Disconnect from MIPS | Reset | Close

Ta nhận thấy

Cache Hit Rate đã giảm xuống, Miss count đã tăng lên, cache hit count giảm xuống