

**Hacettepe University**  
**Computer Engineering Department**  
**BBM234 Computer Organization**  
**2017-2018 Spring Term**  
**MIPS Project**

**Assigned on: 14/03/2018**

**Due date : 24/03/2018 (23:59) – You will use submit.**

## **Introduction**

In this project, you will learn how to write and simulate MIPS code.

You can either use QtSpim or MARS MIPS simulator for this project:

Download QtSpim from <http://spimsimulator.sourceforge.net/> and install it to your computer.

There are several tutorials for QtSpim:

<https://www.youtube.com/watch?v=r8WcV7AiLXs>

<https://www.lri.fr/~de/OtSpim-Tutorial.pdf>

<http://logos.cs.uic.edu/366/notes/MIPS%20Quick%20Tutorial.htm>

Short explanation how to write and simulate your code:

- You should use a text editor such as notepad or wordpad and write your MIPS program first.
- Save your program with an extension .asm.
- Open QtSpim
- Load your program by selecting from the menu.
- Run your program.
- Your results will be either in registers or in memory based on your program output.

The second simulator you can use is MARS, which can be downloaded from the following link:

<http://courses.missouristate.edu/kenvollmar/mars/>

In this homework, you will write and simulate two different programs. You should write comments to explain the purpose of the instructions in your program.

## QUESTIONS:

---

### 1) Arrays using for loops

Write a MIPS code for the following C code fragment. In your code, you are not allowed to use multiplication instructions (mult or mul).

```
int A[4];
int i;
int diff;
for(i=0; i<3; i++)
{
    diff = A[i+1] - A[i];
    if (diff > 0)
        A[i] = 5*A[i];
    else
        A[i+1] = -5*A[i];
}
```

Your array size will be 4. Test your code for the test cases given below.

Test 1: A={2,4,6,8}

Test 2: A={8,6,4,2}

Test 3: A={2,2,6,4}

For each test, save the screenshot of the memory before running the code and screenshot of the registers after running the code.

You can define your array in your code as follows:

```
.data
A: .word 2, 4, 6, 8

.text

main:
    la $t1, A
```

After `la` instruction, the address of the first element of array A will be stored in register t1. Then, you can write your code afterwards. You can also define array size in data segment as well.

---

## 2) Function calls:

Write a MIPS code for the following C code fragment. In your code, you are allowed to use multiplication instructions (mult or mul).

```
int main() {
    int a;
    int b;
    int result = 0;

    if(a == b)
        result = 8*(a + b);
    else
        result = compare(a, b);
    return result;
}

int compare(int a, int b)
{
    if(a<b)
        return punish(a, b);
    else
        return award(a, b);
}

int punish(int a, int b)
{ return (a-b)*2;}

int award(int a, int b)
{ return (a+b)*4;}
```

Test your programs for the following input values:

Test 1: a=3, b=3

Test 2: a=3, b=5

Test 3: a=5, b=3

For each test, save the screenshot of the registers before running the code and screenshot of the registers after running the code. You should clearly show how you use stack in your code.

## **What to Turn In**

Please turn in each of the following items through **submit system**:

1. Assembly code files in .asm file format (array.asm, function.asm). You must add explanation of the instructions as comments.
2. Your report in .pdf file format. Your report should include the explanation of the problems and your code. It should include the screenshots of your tests. Name your report as yourname\_lastname.pdf

## **Grading:**

Arrays	: 40 points
Function call	: 50 points
Report	: 10 points