

CSE 420: Computer Architecture 1

Summer 2017

Project 1

Due Date: Friday, June 02, 2017, 11:59 PM

Objective: Learn to program in MIPS assembly language

In this project, you will be required to write assembly code to implement the following four problems. In all the programs, you are required to strictly adhere to the register conventions, as stipulated by the MIPS architecture. Please refer to the lecture slides, or the ISA reference manual, for the same. [Please read the Submission Instructions section carefully.](#)

Problem 1: String handling [10 points]

Declare a string in the data section:

.data

string: .asciiz "WELCOME TO COMPUTER ARCHITECTURE CLASS!"

Write a program that converts the upper-case strings to lower-case string. To convert an upper-case character to lower case, you can add **0x20** to that character in the string.

Problem 2: Arithmetic expressions [20 points]

Write a program to evaluate the following function in u and v:

$$3u^2 + 7uv - v^2 + 1$$

Here, the variables u and v, are user inputs, and the program should receive them from the user at run-time. Then it should print the outcome computed as per given arithmetic expression. [To understand how to take the user input or write on console, please refer to the sample MIPS example programs].

You are required to create and use two subroutines for this program:

Pay attention to the registers used for passing arguments to *subroutine*, and also the registers used for returning the output from a *subroutine*.

- 1) int Square (a): Return a^2 .
- 2) int Multiply (a, b): Return $a * b$.

Problem 3: Pointers [30 points]

Write a program in MIPS assembly language that will compute the sum of all the elements in an array. Write this program using a function “updateSum,” that takes two parameters; a pointer to the running total, and a pointer to the current element. (To get help on how to deal with the pointers in MIPS assembly, refer to the sample MIPS examples.)

The “C” program looks like this:

```
int sum = 0; int *sumPtr = &sum;

int array[10];

void updateSum(int *total, int *element)
{
    *total += *element;
}

int main()
{
    for (int i=0; i<10; i++)
    {
        array[i] = 3(i+1);
    }
    for (int i=0; i<10; i++)
    {
        updateSum(sumPtr, array+i);
    }
    printf("Sum = %d", sum);
}
```

Problem 4: Recursion [40 points]

Write a program in MIPS assembly language to find `compute(i, x)`, where `compute(i, x)` is defined recursively as:

```
int compute (int i, int x)
{
    if (x>0)
        return compute(i, x-1) + 1;
    else if (i>0)
        return compute(i-1, i-1) + 5;
    else
        return 1;
}
```

Note: Your program should print on the console the computed value `compute(i,x)`. The values for variables `i` and `x` are user inputs to the program. So, your MIPS program should allow the user to enter values for variables `i` and `x` at run-time.

Submission Instructions

- 1) Please make one assembly file for each of the given problems. Save your program files with .asm extension.
- 2) Zip all the files into one file and submit it to the blackboard. Name the zip file Project1-Your10Digit AsuId.zip. In case where you are doing project in a group, only one group member should make submission through blackboard.