

ICS 233, Term 141

Computer Architecture & Assembly Language

Programming Assignment# 2 Due date: Sunday, Nov. 9, 2014

- Q.1.** Write a MIPS assembly language program that implements the following:
- (i) A procedure, PrintA, that prints the content of an array in a two-dimensional format (row-wise) leaving a space between elements. Assume that the array receives as parameters the address of the array in register \$a0, the number of rows in register \$a1, the number of columns in register \$a2, the size of each element in the array in register \$a3.
 - (ii) A procedure, ExchangeR, that exchanges the content of two rows in the array and prints the array after the exchange. Assume that the array receives as parameters the address of the array in register \$a0, the number of rows in register \$a1, the number of columns in register \$a2, the size of each element in the array in register \$a3. Assume that the two row numbers to be exchanged are passed as arguments in the stack.
 - (iii) A procedure, ExchangeC, that exchanges the content of two columns in the array and prints the array after the exchange. Assume that the array receives as parameters the address of the array in register \$a0, the number of rows in register \$a1, the number of columns in register \$a2, the size of each element in the array in register \$a3. Assume that the two column numbers to be exchanged are passed as arguments in the stack.
 - (iv) Ask the user to enter number of rows, R, and read it.
 - (v) Ask the user to enter number of columns, C, and read it.
 - (vi) Ask the user to enter the type of elements to be read and read it. Assume that i indicates integers (words) and anything else indicates characters.
 - (vii) Ask the user to enter an RxC matrix of the required type and read it.
 - (viii) Print a menu from which the user can select one of the following options:
 - 1. Print the entered array
 - 2. Exchange two rows
 - 3. Exchange two columns
 - 4. Exit the program

Note that row numbers and column numbers are assumed to start from 0. Furthermore, note that to multiply two registers \$s2 and \$s3 and store the result in register \$s1, you can use the instruction **mul \$s1, \$s2, \$s3**.

A sample execution of the program is shown below:

Enter number of rows:2
Enter number of columns:3
Enter type of element:i
Enter an array of 2x3 integers:
1
2
3
4
5
6

Select one of the following functions:

1. Print the entered array
2. Exchange two rows
3. Exchange two columns
4. Exit the program

If the user selects the first option, then the following should be displayed:

Array of 2x3 integers is:
1 2 3
4 5 6

If the user selects the second option, then the following should be displayed:

Enter the first row number: 0
Enter the second row number: 1

The array after exchanging rows 0 and 1
4 5 6
1 2 3

If the user selects the third option, then the following should be displayed:

Enter the first column number: 0
Enter the second column number: 1

The array after exchanging columns 0 and 1
2 1 3
5 4 6

If any of the entered row numbers or column numbers are out of range, your program should display an error message and asks the user to reenter the required information.

This assignment can be done by a group of two students. Every group of two students submit only one solution. The solution should be well organized and your program should be well documented. Submit a soft copy of your solution in a zip file. The name of the zip file should be your ID with the new format (i.e. 200157690). Your solution should be submitted in a word file that contains the following items:

- i) Your name and ID*
- ii) Assignment number*
- iii) Problem statement*
- iv) Your solution along with the code*
- v) Discussion of what worked and what did not work in your program. Include snapshots that demonstrate the working parts of your program. If things did not work and you attempted to solve them, mention that and write about the difficulty that you have faced.*