

## 22C:060: Computer Organization

### Homework 2

Total points = 50

Due Tuesday October 1, 2013, **5:00 AM (not PM)**

(Ideally you should finish the work on the previous night and submit it by midnight. This way you can avoid last minute hassles, like network failures or “ICON was not available” kind of problems. **Late submissions will not be accepted**)

1. **Do not consult others. You must solve the problems on your own.**
2. Be generous about using comments to improve readability. This includes a comment at the beginning specifying the purpose of the program.
3. To submit the program, *zip* (or *tar*) them into a single file that has your last name as the prefix. Use ICON drop box to submit your assignment.

### The Questions

**Part 1** (20 points) Write a program using MIPS assembly language to multiply two 8-bit unsigned integers  $x$  and  $y$ . For each integer as well as for the product, use a 32-bit representation. Since the integers are small, there should be no problem with overflow. *Use repeated addition to carry out multiplication.* The algorithm is trivially simple, and does not need any explanation. The user should be able to enter a number between 0 and 255 after the prompts “Enter  $x$ ” and “Enter  $y$ ” are displayed on the screen. The result will be displayed on the screen as “Product = “

*(Do not use the **mult** instruction of MIPS for doing any part of this assignment)*

**Part 2** (15 points) Use a part of the program in Part 1 as a subroutine to compute the product of the elements of two unsigned integer arrays  $A$  and  $B$ , each of size 8. Enter the following array elements for  $A$  and  $B$  directly into the data section:

$A = [9, 13, 10, 20, 1, 6, 9, 14]$        $B = [41, 3, 5, 7, 19, 2, 1, 7]$

You can declare the array elements into the data section of your program as follows.

```
array1:  .byte 10, 15          # create a 2-element integer array (here each element is a byte)
                                     # with its elements initialized to 10 and 15

array3:  .space 40            # allocate 40 consecutive bytes, with storage uninitialized.
                                     # You can use it as a 40-element character array, or a 10-element
                                     # integer array; a comment should explain your preference
```

Show the result as "Product ="

*(Do not use the **mult** instruction of MIPS for doing any part of this assignment)*

**Part 3** (15 points) Write a program that accepts as inputs an asciiz string consisting of a sequence of words separated by one or more blanks, and outputs the string with all blanks removed. Example:

```
input: "the quick brown fox"
output: "thequickbrownfox"
```

The user should be able to enter any string of maximum length 64 characters after the prompt "Enter the input string:" and the result will be displayed on the screen as "The output string is:"