

Project 2

25 points

Project Description

You will learn to code procedures and follow the MIPS register conventions for this project. There will be three procedures/functions in your code: the `main()` procedure, the `revCase()` procedure and the `findMin()` function. Each must have the defined parameters and functionality as explained in these specifications. You are given a starter file `revCaseMin.asm` to add your code to; some of the system calls for output and input are already done.

Write an assembly program `revCaseMin.asm` where:

1. The `main()` procedure
 - a. prompts the user to enter 30 characters and stores these characters as a character array into memory
 - b. invokes the `revCase()` procedure that accepts as an argument the base address of this character array and the number of characters in this array
2. The `revCase()` procedure
 - a. Has two parameters, the base address of a character array and the number of characters in this array. You may not hard code the argument value 30 within this procedure but instead use the 2nd parameter of the procedure. Recall registers `$a0` and `$a1` will be populated by the calling procedure `main()`.
 - b. Calculates the reverse case of the characters entered by the user placing them in a character array and prints the characters in reverse case **using a loop**.
 - c. Invokes the function `findMin()` passing in the required arguments and uses the return value from `findMin()`. The return value from `findMin()` is the minimum character entered by the user after the character string is reversed in case. See details that follow for `findMin()`.
 - d. Prints the minimum character returned from the function `findMin()`
3. The `findMin()` function has two parameters: the first parameter is the base address of the character array it will examine and the second is the number of characters that it will examine to find the minimum ASCII character. `findMin()` returns in `$v0` the minimum ASCII character from the string it examined. You may not hard code the value 30 but must instead use the value in the first argument register `$a1` to aid in the looping structure for finding the minimum character.
4. All register conventions and procedure invocation conventions must be adhered--review the MIPS reference sheet for preserving registers across procedure calls and these conventions.
5. Use the starter code file, `revCaseMin.asm`, making the necessary modifications for this project. Remember, to place YOUR name at the top of the code.

An example dialog follows:

```
This is Dianne Foreback presenting revCase.  
Please enter 30 characters (upper/lower case mixed):  
aBcDeFgHiJkLmNoPqRsTuVwXyZAbCd  
Your string in reverse case is: AbCdEfGhIjKlMnOpQrStUvWxYzaBcD  
The min ASCII character after reversal is: A
```

Project Submission and Deliverables

Submit your code to your Subversion repository. You will need to create a folder `Project2` to include your files for this project.

Please submit the following two files in your Subversion account under the `Project2` folder. Make certain that your assembly code is properly organized (indented, commented, contains your name at the top of the code).

1. The **MIPS Assembly Code Language file** titled `revCaseMin.asm`
2. The **Report file pdf** with screen prints titled `revCaseMin.pdf`

Your report must include:

1. Your name
2. A list of the assembly code file
3. A brief summary of project implementation
4. Results showing the working code via screen prints
5. The conclusion listing the lessons learned and problems faced

If a portion of your code does not work, explain to potentially receive partial credit.

Verify that your report and code is submitted in subversion by viewing from the web browser.

If your report does not load, you will lose points.

Subversion instructions for Project 2. Please read the Subversion Instructions if more help is needed. Recall, Unix is case sensitive, thus, enter the commands exactly as given.

You will need to check out your project directory tree from the repository, add the directory for the new project to the local copy of the tree, inform subversion that new files are under subversion control and then commit the changes back to the subversion repository. The sequence of commands is as follows. The command is in Courier New font with the explanation italicized below the command. Replace **username** with your Flashline username. If prompted for credentials/password, use your Flashline userid and password.

CS 35101 Computer Architecture

```
cd
cd cs35101
```

Entering these two commands in sequence changes the current working directory to the folder cs35101 under your home directory. If you placed the cs35101 directory elsewhere, make necessary changes when entering the commands.

```
svn mkdir https://svn.cs.kent.edu/courses/cs35101/svn/username/Project2 -m "create dir"
```

Creates the folder Project2 in Subversion.

```
svn checkout https://svn.cs.kent.edu/courses/cs35101/svn/username/Project2
```

Inform Subversion that you are checking out the Project1 to your current working directory on wasp or hornet.

In the *Secure File Transfer* window, copy your deliverables, revCaseMin.asm and revCaseMin.pdf, from your personal computer to wasp or hornet in the Project2 folder. Make certain to click the refresh button on the server pane in SSH and maneuver to the correct directory cs35101/Project2 in the server pane. Again, review the Subversion Instructions from Project 1 if needed.

```
svn add Project2/revCaseMin.asm
svn commit Project2/revCaseMin.asm -m "committing"
svn add Project2/revCaseMin.pdf
svn commit Project2/revCaseMin.pdf -m "committing"
```

Copies your deliverables to Subversion and commit.

From a web browser such as Chrome or Firefox (or any other web browser of your choice), go to the URL <https://svn.cs.kent.edu/courses/cs35101/svn/username> to verify your revCaseMin.pdf file and revCaseMin.asm files are submitted and can be opened. If prompted for credentials, enter your Flashline userid and password.

```
rm -rf Project1
```

After verifying on the web that your submission is correct, you can remove the local copy of the project from your Linux account if you so wish with the above command in SSH.

Aside: The following command might come handy. It deletes a directory in the repository (enter the entire command on one line in SSH).

```
svn rm http://svn.cs.kent.edu/courses/cs35101/svn/username/Wrong_Directory -m "deleting Wrong_Directory"
```

Grading

- Working Code (90%)
- Report including results (10%)

If your code does not assemble, a minimum of 50% will be deducted from your score.

If your code does not adhere to the specifications, a minimum of 80% will be deducted from your score.

Reference Materials

- Patterson and Hennessy: Chapters 2.1–2.4, 2.6–2.10, Chapter 3.1–3.3, Appendix A.6 and A.10
- Code Examples (`revString_lb_lbu.asm` and `sum0-100.asm` are good starting points for code review)
- Subversion Instructions (for how to set up and use Subversion)