

FORMAN CHRISTIAN COLLEGE

(A CHARTERED UNIVERSITY)

COMPILER CONSTRUCTION

Programming Assignment 2

It's an open books and open notes assignment. Use of Internet is allowed. This assignment should be done in complete isolation. No group formation is allowed. You CANNOT share your code with any other group. Any such attempt will be dealt with seriously.

Grading Criteria

Working Code: 80%

Properly formatted Report: 20%

Important: You need to submit a well formatted and well written report for this assignment. The report should carry following sections:

- Introduction about the problem in hand especially well written information about the preprocessor and its functions.
- Your code followed by a detailed description explaining how you built up the logic of the program. Additional functionalities and / or exclusions (if any) should be stated with separate heading in bold face font.
- Code description should be accompanied by the screen shots of your output with at least three separate (not large) programs.
- Start early. **NO** additional time in any case what so ever will be granted.
- Viva will be conducted for this assignment. Date will be announced later in class.

Hard Deadline: Report along with the code file/s should be submitted on Moodle course page on or before Monday Jun 13, 2022 before 11:59 pm.

Make sure to create a zip file of your submission and give it a name using given format:

COMP451_A_RollNumber_Assgn-2

Submissions through email will NOT be considered for grading.

Assignment Task [80 Marks]

In this task we will write an assembler. Your program should accept a text file from command line. The file should carry a valid and correct MIPS assembly program. For this assignment we assume that we have a limited number of MIPS assembly instructions. These instructions are:

- add
- and
- or
- andi
- ori
- addi
- sw
- lw
- slt

You may need to refer to the Internet to get yourself acquainted with the instruction formats of MIPS assembly language.

For your convenience, few links are given below:

https://www.dcc.fc.up.pt/~ricroc/aulas/1920/ac/apontamentos/P04_encoding_mips_instructions.pdf

<https://max.cs.kzoo.edu/cs230/Resources/MIPS/MachineXL/InstructionFormats.html>

<https://www.eecs.harvard.edu/~cs161/notes/mips-part-1.pdf>

https://www.dcc.fc.up.pt/~ricroc/aulas/1920/ac/apontamentos/P04_encoding_mips_instructions.pdf

Your task is to write a program in C that accepts a text file containing few lines (minimum three) of MIPS assembly program using the above instruction set. (It does not matter whether or not the program makes some useful output).

Your program should then read the file line by line and decode the instruction into its equivalent machine code.

Note that you are not supposed to generate the output of the program. Your only intention is to generate the equivalent machine program.

Note: We can provide any tailor made program (using above instruction set only) to test your code.

COMP 451

The output of the program should be as shown:

Sample Run 1

```
$ gcc assgn2.c -o assgn
```

```
$/assgn inputFile
```

Assembly language program:

```
add    $9,$10,$11
```

```
add    $12,$9,$10
```

```
and     $13,$10,$12
```

It should then display the machine code of each instruction individually in binary format.

(Tentative) Machine Code:

```
0000000010100101101001000000100000
```

```
0000000010010101001100000000100000
```

```
0000000010010110001101000000100100
```

Sample Run 2

```
$ gcc assgn2.c -o assgn
```

```
$/assgn inputFile
```

Assembly language program:

```
add    $t1,$t2,$t3
```

```
sw     $t1,($t2)
```

```
addi   $t4,$t1,4
```

```
lw     $t3,16($t2)
```

```
and     $t5,$t2,$t4
```

(Tentative) Machine Code:

```
0000000010100101101001000000100000
```

```
10101101010010010000000000000000
```

```
0010000010010110000000000000000100
```

```
100011010100101100000000000010000
```

```
0000000010100110001101000000100100
```