

Task 1:

Computer Organization and Assembly Language CMP 223

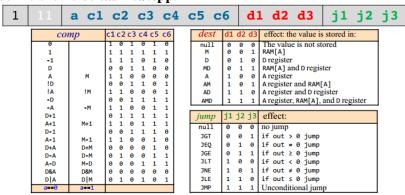
Term Project (Spring 2021)

Total Marks: 100

[50]

Write down a Hack Assembler program that opens and reads a text file (with .asm extension) containing valid Hack assembly instructions (having no symbols) and each instruction separated by a new line character. The hack assembler generates a text output file containing the corresponding 16-bit machine code for each instruction (with .hack extension). Hack assembler program performs translation with following characteristics:

- Ignore all the whitespaces, Empty lines /indentation, Line and In-line comments.
- Assume that no builtin symbols, labels and variables are being used (Break symbolic instruction into its underlying fields).
- For each instruction use A-Instruction and C-Instruction
 - A-Instruction: translate the decimal value into a binary value.
 - C-instruction: for each field in the instruction, generate the corresponding binary code.
 - Consider the table given below to verify the binary code against C-instructions
 - Combines the binary codes into 16-bit instructions.
- Write the translated instruction in the output file with the same input file name but with .hack extension
- The name of the code file should be **task1.c/.cpp**.



<u>Task 2:</u> [50]

Extend the Hack Assembler program written in Task-1 that can handle the symbols as well. Hack assembler program performs translation with following characteristics:

- Hack Assembler uses the Two Pass approach to translate an assembly program.
- In the first pass, it creates an empty Symbol Table and initializes it with symbols and their values.

- Assembly program contains symbols of 3 types:
 - **Pre-Defined Symbols:** Start with "@" and occur only in A-instructions.
 - Initialize the table with 23 pre-defined symbols.
 - Consider the given table for 23 pre-defined symbol values.
 - Label Symbols: Use to label destination of goto command
 - Label declarations are not translated and generate no code.
 - Replace the label symbol with the address of memory location holding the next instruction.

Symbol

RO

R1

R2

R15

SCREEN

KBD

SP

LCL

ARG

THIS

THAT

Value

0

1

2

15

16384

24576

0

1

2

3

4

- Read the source file and look for label declaration only, and on encountering a label declaration, enter the label name with its corresponding address in the symbol table
- In the second pass. It replaces the variable symbols with their corresponding values
 - Variable Symbols: Any symbol appearing in the program which is not predefined and not used to refer to goto commands.
 - If seen for the first time, assign a unique memory address starting from 16
 - Replace symbol with the address in the symbol table
 - For A-Instruction translate the decimal value into a binary value.
 - For C-instructions consider the table given in Task1.
- It handles White Spaces.
- Write the translated instructions to the output file with the name same as the input assembly file but with .hack extension.
- The name of the code file should be **task2.c/.cpp**.

Submission Instructions:

- Solutions to all the parts must be your own hard work. DON'T let anyone copy your assignment. In case of a copy both students will be awarded a ZERO may be some negative marks as well.
- If you have copied even a single line from the internet/webpage, mention it in your file using a precise comment in the task description. Moreover you must have a complete understanding of it.
- You may implement your Hack Assembler programs using C/C++.
- Deadline for submission of the Term Project is Wednesday, May 19, 2021 till 11:59 pm.
- The title of your assignment folder should be rollno_TermProject containing the above mentioned program files task1.c/task1.cpp and task2.c/task2.cpp along with output files. Push the folder on your Bitbucket repository before the deadline.
- Late submissions will NOT be accepted. So start doing the problems from today so that you can submit your assignment in time.



TIME IS JUST LIKE MONEY.
THE LESS WE HAVE IT;
THE MORE WISELY WE SPEND IT.
Manage your time and Good Luck