CSCI 3150 Assignment Report

ZHANG Jingze (SID: 1155107857)

1.For the conveniences and difficulties of C and COBOL, we can consider them in different aspects. For the "reading file in certain format", COBOL is more convenient with its additional support for reading data in format. C in comparison allows us to read the file flexibly and could not read the file in format with the given function. The defined format for reading file data and the structure for reading file function is shown with the following code examples.

```
           FD INS-FILE.
           01 INSTRUCTOR.
               03 INS-COURSE-ID PIC X(5).
               03 REQ-SKILLS PIC X(15) OCCURS 3 TIMES.
               03 OPT-SKILLS PIC X(15) OCCURS 5 TIMES.
```

```
int ins_readline(FILE* insfp, char* insline){ // basic structure for the read line function
  // define variables idx and c
  // read the line character by character for the line with c and idx
  // here: parse the line read from file and get the data
}
```

For the "simulating loops", C is more convenient. Unlike COBOL, C provides structual loop syntax such as "for" and "while" while COBOL only provides the primitive jump commands like "go to". For the "procedure and function call" part, the C also prevails over COBOL because while COBOL only has "code block" that you can use as a subroutine for a certain function, the C allows you to distinctly define a "function" that can handle problems with itself.

2.Compared to Python, the COBOL is even more primitive and more inflexible. COBOL could only handle problems with relatively simple logic while Python allows us for more usage with its universal support for various functions and its flexibility in syntax. Moreover, the object-oriented support of Python allows us to store more information and handle more problems for a certain data collection. The dynamic typing feature of Python makes it easier to store the data conveniently and change the content according to various requirements.

3.For the the application in this assignment, COBOL is not very suitable. Because for this assignment, we need to check all the candidate information for each course code. With more course code to process, we need to read the candidate file more times and thus wastes more time for our program. In this way, the efficiency of the program is greatly limited. Moreover, we need a 2-layer loop for the whole program and the poor support of loop structure of COBOL also makes it more difficult to form the code.

4.In the COBOL code, we use a 2-layer loop structure for the whole program. For eachi layer, we construct it in a spearate submodule as a code block and use the "go to" command to transfer between those modules. In the first module, we read the course code line by line in a loop and deal with each code one by one. For the second module, we apply a loop on the candidate line by line and find the top 3 candidates according to the score.

In the C code, we construct the code with the main module and other modules as functions called by the main module. In the modules called by the main module, we implement the functions of reading the candidate file line by line and reading the course code file (instructor file) line by line. We also implement some commonly used function within such modules such as "move values between variables" and "compare value of variables".