1. There are eight categories outlined for application software on the content page of this activity. Make a list outlining what application software is available on your computer for each of the eight categories?
    a. Word Processors:
        i. Microsoft Word
        ii. NotePad
        iii. WordPad
    b. Spreadsheets:
        i. Microsoft Excel
    c. Presentations:
        i. Microsoft Powerpoint
    d. Databases:
        i. File Explorer
        ii. Partition Manager
        iii. Windows Explorer
    e. Graphics:
        i. Paint
        ii. Paint 3D
        iii. Adobe Suite (Photoshop, After Effects)
    f. Games:
        i. Solitaire
    g. Communications:
        i. SSH Client
        ii. TeamViewer
        iii. FTP Client
        iv. Remote Desktop Connection
    h. Browsers:
        i. Google Chrome
        ii. Internet Explorer

        iii. Microsoft Edge

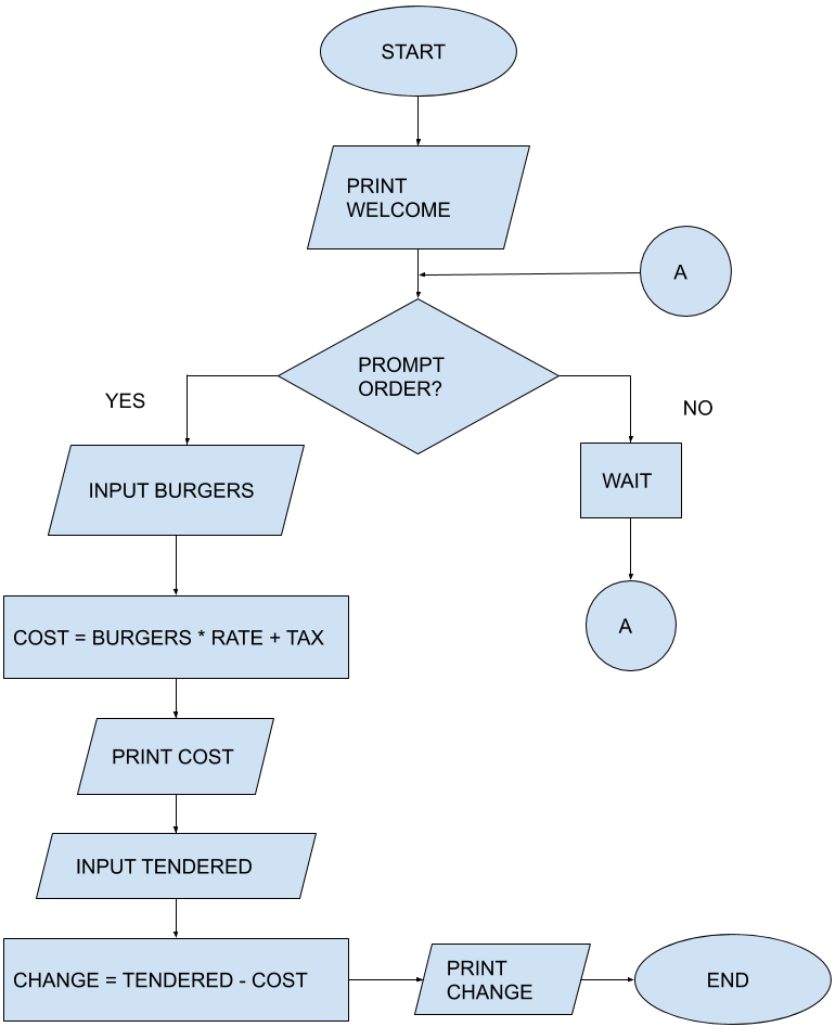
2. List and explain the different stages of the Software Development Life Cycle
    a. Analysis:
        i. The first stage is to identify the problem and create an overall view of all the elements to creating a solution for the problem. An overview of how the program should run is developed.
        ii. The overview of how the program functions include identifying the user, what form factor they will use the program on, how the program UI will work, the functions of the program, and the parameters it takes in, outputs & stores.
    b. Design & Development (Develop the Algorithm):
        i. The second stage is where the actual algorithm is designed, all the details of how the system will function are laid out and planned. The program is broken down into parts to develop.
        ii. The algorithm design can be plotted and planned using a flowchart, pseudocode, or even IPO chart
    c. Implementation (Code the Solution):
        i. The third stage is often the longest, code is written in parts as specified by the algorithm design. The entire coding process follows alongside the Algorithm Design from the previous stage.
        ii. Additionally, as code is written, proper documentation correlating line by line must be written for other programmers to understand and further develop the code.
    d. Testing/Verification and Deployment:
        i. The fourth stage is the testing stage. Varying data is inputted into the program and run through to test the program functionality. A trial of the app is often completed within a small group to receive feedback and discover any errors or flaws. The program is then

rectified according to the feedback and released to the client/target audience.

e. Maintenance:
    i. After the release of a program, feedback is received from the target audience. The program is modified according to the feedback and released in the form of an Update in. Updates are released every set time duration.

3. Explain the difference between source code and machine code
    a. Source code is a set of instructions written in a certain programming language, with instruction for the computer to complete certain actions. The programming languages are written in English like format with their own respective formats, examples include Java, C++, and Pascal amongst others which are high-level languages.
    b. Machine code is a set of instructions written in a low-level programming language called binary. Binary consists of base-2 numbers, 0's & 1's. Binary is the only language a computer can understand, and be actionable upon.
    c. The difference between Source Code & Machine Code, is that source code are instructions written in a more programmer-friendly format, which are then translated using a compiler or interpreter into a harder to understand format which is machine code.
4. Explain the difference between an interpreter and compiler
    a. An interpreter reads the source code of a program line by line, translating and executing the code line by line.
    b. A compiler reads the entire source code and then translates into object code, which is very similar to machine language.
    c. The difference between an interpreter and compiler is that an interpreter reads, translates and executes code line by line, whereas a compiler reads and translates as a whole, then executes.

5. Are there any advantages/disadvantages related to using an interpreted programming language over a compiled language?
    a. Using an interpreted or compiled programming language is solely based on the intention of the program. The advantages to using an interpreted language over a compiled language are smaller executable program size and dynamic typing. Additionally, if any changes are made in an interpreted program there wouldn't have to be an entire rebuild as is the case with a compiled language. Disadvantages to using an interpreted language over a compiled language are execution times, whereas compiled languages generally execute faster than interpreted (Compiled Versus Interpreted Languages, n.d.).
6. Flowchart of ordering a hamburger (as a fast food restaurant order taker)



Pseudocode:

Begin Program

    Print Welcome

    Prompt Order

        Ready to order

            Calculate Cost with taxes

            Print Cost

            Take Amount Tendered

            Calculate Change

            Print Change

        Not ready: Wait and Prompt again

End Program

References

Compiled Versus Interpreted Languages. (n.d.). Retrieved from https://guide.freecodecamp.org/computer-science/compiled-versus-interpreted-languages/