In this unit, we are exploring the idea of a conceptual model. In order to develop effective and efficient programs, we need to develop a design for the program. The worst thing that a programmer could do is to simply sit down and attempt to write a program. Preparation is required. Many of you have have taken a course in writing. You will remember that we are taught techniques such as outlining, or diagramming to build a framework of what we are to write. The basic idea of these techniques is to have a good idea of what you are going to write, the points you will make, the flow of the piece, the conclusions that you must reach BEFORE you pen the first word.

Programming follows a similar process. Before we, as programmers, attempt to write the first line of code, we need to have a pretty good understanding of how the program will be constructed, how it will flow, what processes it will need to perform, and what error checks will be required. We develop this plan, or 'conceptual model' as I like to refer to it, using one of two modeling approaches. In this unit we learned about flowcharts which provide a visual diagrammed approach to describing the operation and flow of a program as well as pseudo code which uses formatted English like statements to describe the operation of the program.

In your discussion response, provide a description of both flowcharts and pseudo code. Discuss the advantages of each and describe any disadvantages that might make you want to use one method over the other.

(minimum word count 75. for the initial post)

(minimum word count 25, for replies)

Pseudocode is a shorthand notation for program code, using regular, understandable, language to describe each step in a program. It is much simpler and easier to understand than code of some other CS languages, and emphasizes the outcome of the code, rather than the technical details and correct syntax1. Pseudocode can easily be completed on Microsoft Word, easily modified, and easily converted to a programming language (vs. flowcharts). However, they are not visual, add an additional level of documentation to maintain, have no accepted standard (unlike flowcharts), and it is more difficult to follow the logic or clearly represent decision points2.

Flowcharts are .. flowcharts. They are graphical structures that shows the logical structure, progression, and decision tree or junctions in the solution to a problem (algorithm). They are used for many industries and fields, but when employed for computer code, there are standards for different shapes, where cylindrical represents start/end points, parallelogram represents input/ouput operations, rectangle represents processes, etc. Flowcharts are allegedly better for smaller concepts and problems, offer an easy method of communication, but may not be as flexible as pseudocode, and are more time consuming to create and modify3. Flowcharts also are visual (advantage), but require special software (disadvantage)4.

1. Pseudocode and Flowcharts. Carbon Mellon Robotics Academy. URL: <http://www.education.rec.ri.cmu.edu/products/cortex_video_trainer/lesson/media_files/hp_pseudo_flow.pdf>. Accessed February 19, 2018.
2. Advantages and Disadvantages of Pseudocode. (n.d.). Retrieved February 19, 2018, from <https://www.scribd.com/doc/244475321/Advantages-and-Disadvantages-of-Pseudocode>
3. Differences Between Psuedocode and Flowcharts. (n.d.). Retrieved February 19, 2018, from <https://www.techwalla.com/articles/differences-between-psuedocode-and-flowcharts>
4. Pseudocode: An Introduction RULES FOR PSEUDOCODE. (n.d.). Retrieved February 19, 2018, from <http://faculty.ccri.edu/mkelly/COMI1150/PseudocodeBasics.pdf>