Assignment 1

Heman Baral

Each question is worth 7 points for a total of 49 points.

1. Explain how von Neumann’s machine model represented an advance in computing?

Von Neumann’s machine was the first to store a program and it was an advancement for the operators of computers because this allowed for the operator to be able to enter in a bunch of codes to make a computer perform the required action, rather than rewiring the whole thing. This made for much quicker adaptation to different tasks.

1. Of the languages discussed during the Early History of Programming Languages lecture, which do you think was the most important historically? Why?

I think FORTRAN was the most important language. It provided the first programming language with a complete compiler, the software needed to translate a high level language humans could use to write software easily into assembly language or machine code computers could understand. On average, programs written in FORTRAN were 20 times shorter than equivalent handwritten assembly code. This language was created to write program to solve scientific and engineering problems. It introduced integer and floating-point variable

1. Explain the difference between a data structure, a data type and an abstract data type.

A data structure is a way to represent the relationship among different available data items appearing in a group and along with set of operation which can be performed on that group. Frequent operations include storage, searching, insertion, deletion and storing. Whereas, datatype is used to declare or define a variable. It specifies the data stored format for the variable. For example, char, int, float, double, etc. are all data types. Abstract data type is a mathematical model of the data object that make up a data type as well as the functions that operate on these objects. It can describe the way components are related to each other for example lists, graph, and array are all Abstract data type.

1. Discuss the role of abstraction when using procedures and functions. How does recursion build in another layer of abstraction?

Abstraction hides the complexity and details of the procedures and functions. So, in a program we won’t have to see all the code written for a procedure or function when calling it. Recursion simplifies the code and allows it to reference itself.

1. Discuss higher-ordered functions. How does abstraction play a role in these functions?

High order functions are functions that operates on the other functions by either taking them as argument or returning them. Abstraction allows us to call many functions without having to write many lines of code to return the same result.

1. Define the von Neumann Bottleneck and explain how it is related to the Processor-Memory performance gap.

The idea of von Neumann bottleneck is that computer system throughput is limited because the ability of processors is less compared to top rates of data transfer. Due to that the processor remains idle for a certain amount of time while memory is accessed.

1. What are the advantages and disadvantages of the use of a virtual machine and translation of the code into byte code?

The advantage of using virtual machine is that you can send your code in byte code to others and they can run it on a virtual machine if you don’t have the right hardware. One disadvantage would be it is not as efficient as being directly converted to assembly.

The following MARIE program is worth 51 points.

Create a MARIE assembly program that calculates and displays the area of a rectangle, given a rectangles height and width, then calculates how many times larger a given area is than the area of the rectangle.

The MARIE program will be prompted to enter the length and width as decimal values; once these values have been read by the program, the area of the rectangle must be calculated and output. The program must then prompt for an area as a decimal value. Once input, the program will calculate and output how many times bigger the given area is than the area of the given rectangle. The program must then properly terminate. For example, if the input is:

4 (Length)

5 (Width)

100 (Area)

Your program should print:

20

5

A few notes:

* “Prompting” the user for input should consist solely of the input instruction being executed. No string should be output to the console to inform the user what data is being requested, rather the first input should always be the length, the second the width, and the third an area.
* When calculating how many times bigger the given area is than the area of the given rectangle, calculate to the nearest whole number. Do not attempt to calculate decimal numbers.
* A given area may be smaller than the calculated area.
* You must comment your program.

When finished, upload your completed .mas file to Blackboard. The .mas file must be named Assignment1.mas and you should only upload your .mas file. Do not upload any of the other files generated by MARIE.

Grading Rubric:

|  |  |
| --- | --- |
| Category | Points |
| Files submitted correctly | 5 |
| Comments used appropriately | 10 |
| Inputs and program format correct | 10 |
| Correct output | 21 |
| Total | 51 |