Austin McCalley

Dr. Jennifer Parham-Mocello

Computer Science 160-020

19 October 2019

Assignment 4

*Create Your Own Plan in MyDegrees*

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

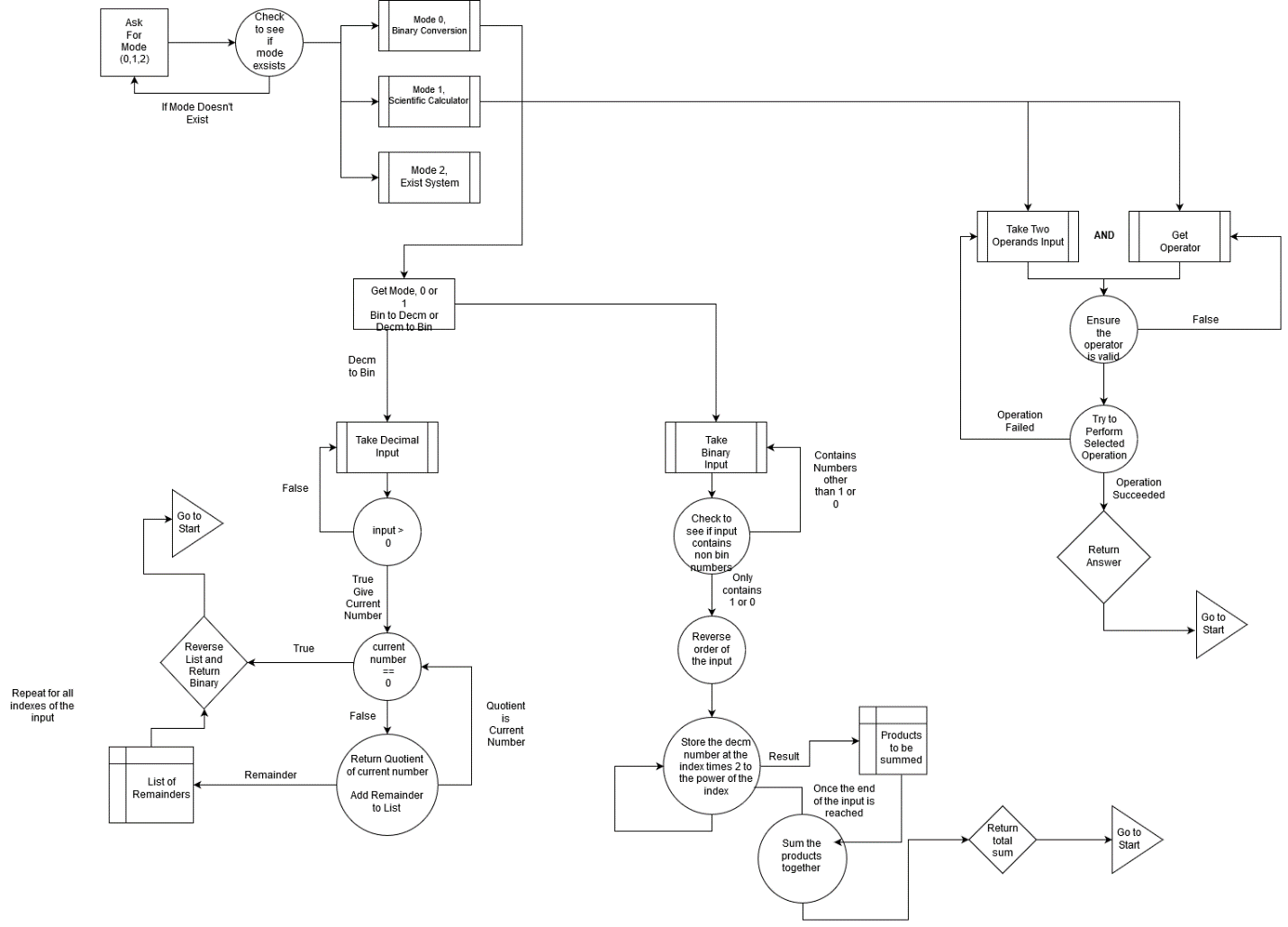
*Step 1: Understanding the Plan*

The goal for this assignment is to build a calculator that has two separate modes, the first mode is a programmer mode which convers any unsigned decimal number and convert it to a binary number for the user. When printing the decimal number in its binary complement number there should be no leading zeros, as well as, if a user inputs a number which is not positive there must be an error number and attempt the problem again. The second mode is a scientific mode which asks for two operands to perform actions on and then an operator. The program must allow for floats and integers, if there is any error or mathematical error there must be an error message. When running the program there must be three possible options, the first two must be to enter either mode or the last mode is to exit the application. After either operation is performed the mode selector must be prompted again.

*Step 2: Devise a Plan*

The application is going to have two different modes, each with their own function. The decimal to binary convertor is going to take an input of the decimal number and then find the most significance bit and then work backwards to find all the bits. Once all the bits are found then you reverse the string and return it to the system. If the input number is negative, then we will return an error and call the function again which allows the function to be recursive if needed. The scientific calculator mode is going to have its own function which will ask for the two operands and the operator. We will return an error and then recursively call the function to ensure that the function runs correctly.

*Flowchart*

**

*Step 4: Looking Back*

The test plan is going to test all inputs for the given modes and ensure that it brings us to the correct function, as well as, we are going to also pass in inputs which aren’t a mode and going to ensure that it gives us an error message and ask for the mode again.

For the binary conversion I checked the inputs of the first 15 decimal numbers and ensure that they result in all the binary numbers from 0 to 1111. The numbers were correct. As well I tried to pass in incredibly complex numbers to ensure that it returns correct binary inputs and then compared them to what they should be using an online decimal to binary converter to ensure that they were right. Lastly, I would pass in a set of negative numbers to ensure that it gave me an error in which it did.

For the scientific option I checked all possible inputs from 0 to 5 going by ¼ numbers with all different operators to ensure that the answers are correct. I then tried ensuring that dividing by 0 gives us an error message as well as using numbers which are over the max integer count to ensure that it gives us an error and asks for the numbers again.