

Program Design & Testing Document for Program 4

Lyell C Read

Problem Statement

The problem asks me to create a program that can fulfil the following requirements: (source: canvas / Justin Goins):

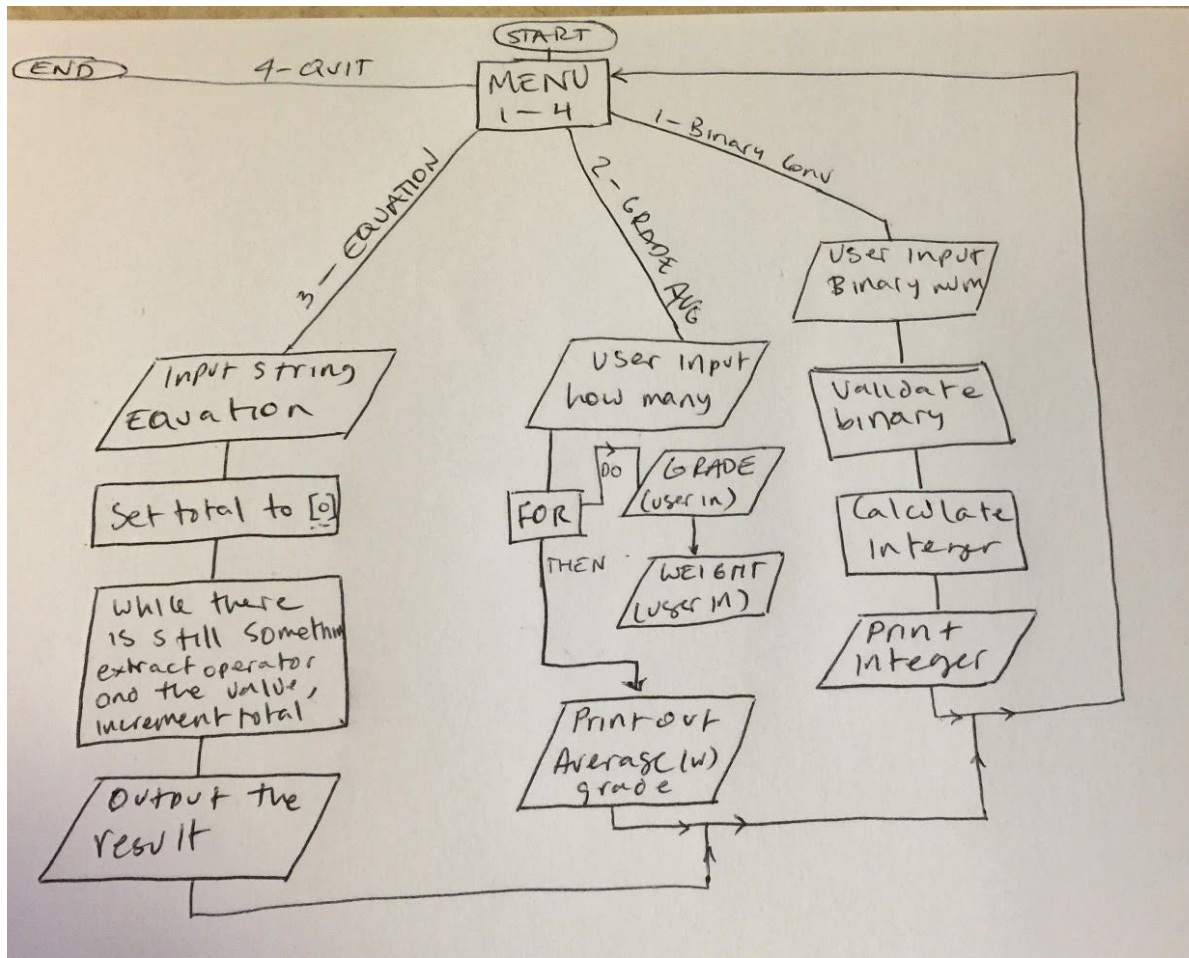
- Clarification (10/23/18): Your code should detect cases where the number or ordering of operands is incorrect. For example, the calculation "6 + 4.125 + 7 9" would display an error, as would "6 + / 6".
- There should be spaces between each operand and operator.
- The input should be taken as a string.
- There should not be competing precedence. Everything will be calculated left to right.
- The binary conversions will only handle unsigned, positive values.
- The grade calculator should prompt for the number of grades to be entered, collect those grades
- from the user and average them. The user should also be given an option to calculate a weighted
- average. For this option the program will need to take a weight from the user and multiply it by
- the average. If you are unfamiliar with weighted averages, see the introduction at: <https://www.wikihow.com/Calculate-Weighted-Average> (Links to an external site.)Links to an external site.
- Your program should handle all input errors such as a user putting in an invalid character or
- number.
- All function bodies must be 15 lines of code or less, including main(). Whitespace and
- lines with just curly braces do not count in the line count.
- No global variables or goto functions.

To summarize the above, the program will have to evaluate and error check an input in the form 4 - 2 / 44 - 2.3 as a string. The program will also convert unsigned binaries, and calculate grades using a weighted average. Also, there will be an option to average the grades in an unweighted manner (could just substitute 1/grade_num for weight).

Understanding the Problem

The problem asks for a program that can convert binary input to integer output, that can evaluate string equations (presented as "9 + 3 / 5") and that can calculate the average grade using a weighted score system. The hardest constraint will be that all functions will have to be under 15 LOCs.

Devising a Plan



Pseudo Code (Simplified)

menu ():

- List 4 options

- Seek user input

- Call appropriate function for user input, or exit() if 4 (see diagram)

Binary to integer:

- For every digit in the binary number,

- check that it is either a 1 or a 0 - otherwise, break

- For every digit starting with the highest index (len-1)

multiply by the appropriate quantity of 2^x
Add these together and print the value,
break

next-operator (eq):

Copy off the first digit of the eq
Remove the first 2 digits of the eq (operator and space)
Return the first digit of the equation which is the operator

Next-integer (eq):

While the value after the first one (index [0]) is an integer:
Flag += 1
Copy off all the digits from 0 --> flag
Remove all the digits from 0-->flag+1 (to account for the space)
Return the number

Evaluate:

Total = next-integer (strips the initial number, and leaves the operator there, while also making a base)

While there's still something left as part of eq:
Operand is the return of next-operator
Integer is the return of next-integer
Depending on what operator is, perform the operation on total and integer
Print total which is the outcome value

Weighted grade calculator (uses mostly doubles)

Ask user for num, which is how many grades they want to enter
grade=0
For num:
If user has selected to do unweighted:
weight=(1/num)
Else
Ask for weight
Ask for grade
Grade += (100*(weight/100 * grade/100))
Print grade

Predicted Results

Value	What Should Happen	Does This Happen
-------	--------------------	------------------

"	Error - nothing entered	yes
"2"	2	yes
"oasdjfasone"	Error - improper formatting	yes
"9 + 9 / 73234"	0	yes
"99 + 11 * 100"	11000	yes
"9/ 2 + 5"	Error - improper formatting	yes
" 9 9 / 4 + 1"	Error - expected an operator	yes