

Assignment 2: Calc and Format readme

CALC:

Calc executable takes arguments in the form as listed in the assignment pdf, with the exception of multiplication. Multiplication requires a “*” instead of * as weird things were happening when just * was input.

Calc can handle addition multiplication and subtraction of inputs whose sum/product/difference will be NO LARGER than 32 bits.

Calc does calculations by:

- Converting all inputs into binary and storing it in a BinNum struct, regardless of base
- Converts the binary into decimal
- Performs the designated operation (+ - *)
- Converts from decimal into designated output base

Calc has a multitude of functions for converting to the various bases, and functions dedicated to converting bases into their string counterparts.

Most of the functions run in $O(n)$ time, as they pass over the entire input string multiple times throughout a single call of calc.

FORMAT:

Format takes in a 32 bit bitstream and converts it to either a float or int. There are two functions outside of main.

CreateIntOut returns a string containing the integer representation of the bitstream.

CreateFltOut returns a string containing the float representation of the bitstream.

Both of the functions run with $O(n)$ time as they iterate through the bitstream multiple times as a part of their algorithms.