

CSC 406: Computer Systems I: 2019 Winter

Assignment #1

Purpose:

To go over:

- C programming basics
- Pointers and addresses

Overview:

We will program the *Babylonian Square Root Method*.

1. Let `number` be the number to get the square root of
2. Start with an `estimate` of 1.
3. Recompute `estimate = 0.5 * (estimate + number/estimate)`
4. Go back to 3

Stop the looping when either of these is true:

- `(estimate*estimate) == number`
- The number of iterations gets to 100, the maximum

1. Please copy-and-paste the following files (0 Points):

babylonianSqrt.c

```
#include
#include
const int TEXT_LEN      = 64;
void      obtainFloat (float* fPtr)
{
    char    text[TEXT_LEN];

    // YOUR CODE HERE

}

float      squareRoot    (float number,
                          int    maxIters,
                          int*   numItersPtr
                          )
{
    float estimate      = 1.0;

    // YOUR CODE HERE

    return(estimate);
}

int      main            ()
{
    float f;
    float ans;
    int    numIters;

    obtainFloat(&f);
    ans      = squareRoot(f,100,&numIters);
    printf("squareRoot(%g) approx. equals %g (found in %d
iterations).\n",
           f,ans,numIters
    );
    return(EXIT_SUCCESS);
}
```

2. Finish obtainFloat() (40 Points):

It should ask the user to enter a number from 0 to 65535, and lets the user enter a number. If the user enters a number outside that range, it asks again. The number is **not** returned, but number is placed in the address to which fPtr points.

3. Finish squareRoot() (60 Points):

After setting estimate to 1.0, it should loop and recompute estimate with the expression above. Every time it loops, it should increment the integer to which numItersPtr points. The loop should stop when the condition given above is met.

Sample Output:

```
[instructor@localhost Assign1]$ ./babylonianSqrt
Please enter a floating point number (0 - 65535): -7
Please enter a floating point number (0 - 65535): 1000000
Please enter a floating point number (0 - 65535): 9
squareRoot(9) approx. equals 3 (found in 5 iterations).
[instructor@localhost Assign1]$ ./babylonianSqrt
Please enter a floating point number (0 - 65535): 100
squareRoot(100) approx. equals 10 (found in 7 iterations).
[instructor@localhost Assign1]$ ./babylonianSqrt
Please enter a floating point number (0 - 65535): 101
squareRoot(101) approx. equals 10.0499 (found in 100 iterations).
[instructor@localhost Assign1]$ ./babylonianSqrt
Please enter a floating point number (0 - 65535): .25
squareRoot(0.25) approx. equals 0.5 (found in 4 iterations).
```