Problem 1: Practice with Loops

Below are four code segments, including a variable initialization and a loop. Write the value that the requested variable contains after the loop has completed.

```c
int n, m;
n = 1;
for ( m = 1; m <= 5; m++ )
{
    n = n*m;
}

int a, b;
a = 5;
while ( a < 100 )
{
    b = a+2;
    a = b * a;
}
```

What is $n$? _____

What is $y$? _____

What is $b$? _____

What is $f$? _____

Problem 2: Programming Loops

Download `loops.zip` from the lecture outline. Expand it into your Z: drive and open `loops.c`. There are blanks in the implementations of the functions `squares` and `exponents` where you need to insert a loop structure.

**ONLY WRITE CODE WHERE INSTRUCTED BY THE COMMENTS.**

- **squares** takes an integer $n$ and outputs all the squares $1^2, 2^2, 3^2, \ldots, n^2$. The output code is written, but the loop is missing. Hint: Use a `for` loop and calculate the square of the conditional variable.

- **exponents** takes a decimal $b$ and an integer $m$ and outputs all powers of $b$ from 0 to $m$: $b^0, b^1, b^2, \ldots, b^m$. The output code is written, but the loop is missing. Hint: keep a counter of the current power, and use the `pow` function.

To check your work, type the command `testLoops`. This will run your program with the input 20 2.5 5. Show the instructor the output. (Hint: if `testLoops` outputs nothing, then you did it right!)

**Bonus:** Write the `exponents` function without using the `pow` function. Show the instructor your code if you have done the bonus.