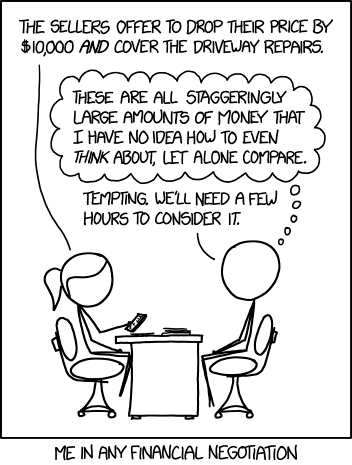
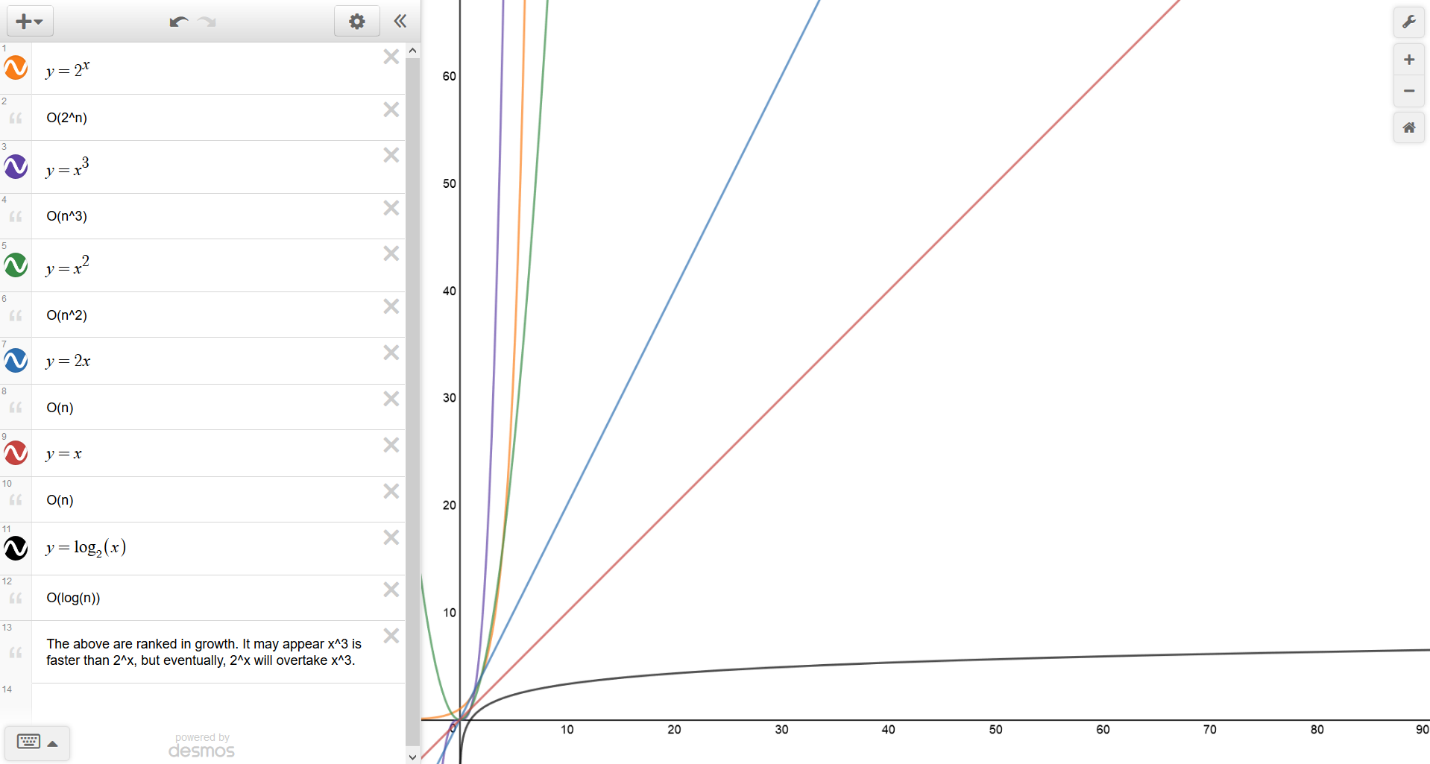
**CISP 430**

**Lawrence Wong**

**HW 1**



**Math Functions**

**Curves:**

**Foo 1**

**Description:**

This program is an instructive implementation of a program that has time complexity of O(n).

**Source Code:**

/\*

@author: Lawrence Wong

@version: 1/31/18

\*/

#include <iostream>

#include <fstream>

using namespace std;

int foo1(int n)

{

int counter = 0;

for (int i = 0; i < n; i++)

counter++;

return counter;

}

int main()

{

ofstream fout("foo1.csv");

fout << "n,return value" << "\n";

for (int i = 0; i <= 16; i++)

{

fout << i << "," << foo1(i) << "\n";

}

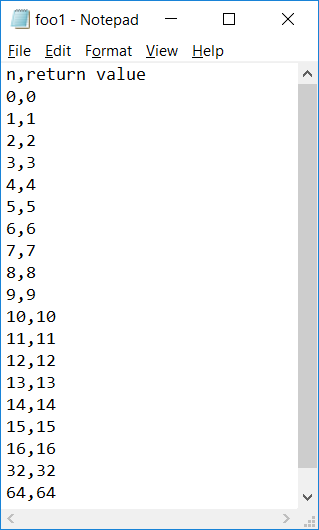
fout << 32 << "," << foo1(32) << "\n";

fout << 64 << "," << foo1(64) << "\n";

return 0;

}

**Output:**



**Graph (Rank 3):**

**Foo 2**

**Description:**

This program is an instructive implementation of a program that has time complexity of O(n^2).

**Source Code:**

/\*

@author: Lawrence Wong

@version: 1/31/18

\*/

#include <iostream>

#include <fstream>

using namespace std;

int foo2(int n)

{

int counter = 0;

for (int i = 0; i < n; i++)

for (int j = 0; j < n; j++)

counter++;

return counter;

}

int main()

{

ofstream fout("foo2.csv");

fout << "n,return value" << "\n";

for (int i = 0; i <= 16; i++)

{

fout << i << "," << foo2(i) << "\n";

}

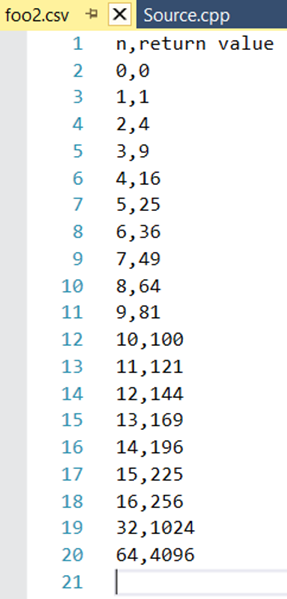
fout << 32 << "," << foo2(32) << "\n";

fout << 64 << "," << foo2(64) << "\n";

return 0;

}

**Output:**

****

**Graph (Rank 2):**

**Foo 3**

**Description:**

This program is an instructive implementation of a program that has time complexity of O(log(n)).

**Source Code:**

/\*

@author: Lawrence Wong

@version: 1/31/18

\*/

#include <iostream>

#include <fstream>

using namespace std;

int foo3(int n)

{

int counter = 0;

for (int i = n; i > 0; i /= 2)

counter++;

return counter;

}

int main()

{

ofstream fout("foo3.csv");

fout << "n,return value" << "\n";

for (int i = 0; i <= 16; i++)

{

fout << i << "," << foo3(i) << "\n";

}

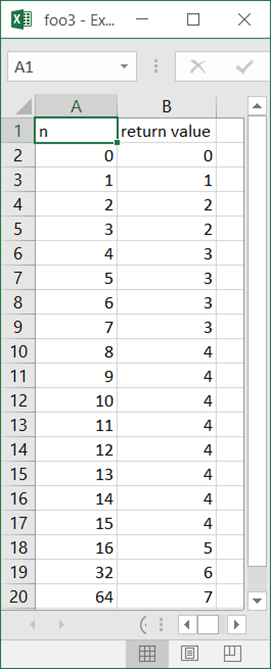
fout << 32 << "," << foo3(32) << "\n";

fout << 64 << "," << foo3(64) << "\n";

return 0;

}

**Output:**



**Graph (Rank 4):**

**Foo 4**

**Description:**

This program is an instructive implementation of a program that is EXTREMELY BAD.

**Source Code:**

/\*

@author: Lawrence Wong

@version: 1/31/18

\*/

#include <iostream>

#include <fstream>

using namespace std;

int foo4(int n)

{

static int counter = 0;

counter++;

if (n > 0)

{

foo4(n - 1);

foo4(n - 1);

}

return counter;

}

int main()

{

ofstream fout("foo4.csv");

fout << "n,return value" << "\n";

for (int i = 0; i <= 16; i++)

{

fout << i << "," << foo4(i) << "\n";

}

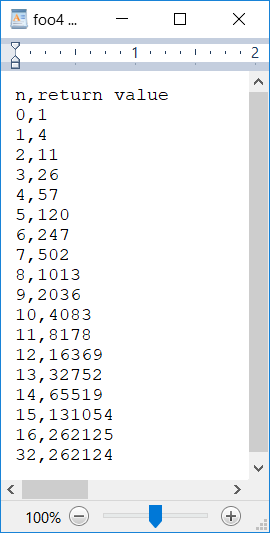
fout << 32 << "," << foo4(32) << "\n";

// fout << 64 << "," << foo4(64) << "\n";

return 0;

}

**Output:**



**Graph (Rank 1):**

**More Info:**

I did not include 32 in the graph as I believe some integer overflowing occurred.

**Arithmetic Series**

**Description:**

This program uses the bad way to find the sum of 1+2+3+…

**Source Code:**

**Output:**

**Analysis:**

This is

**Geometric Series**

**Description:**

This program uses the bad way to find the sum of 1+2+3+…

**Source Code:**

**Output:**

**Analysis:**

This is

**Geometric Series v2**

**Description:**

This program uses the bad way to find the sum of 1+2+3+…

**Source Code:**

**Output:**

**Analysis:**

This is

**Another Series**

**Description:**

This program uses the bad way to find the sum of 1+2+3+…

**Source Code:**

**Output:**

**Analysis:**

This is