// This program uses a static local variable.

/\*#include <iostream>

using namespace std;

void showStatic(); // Function prototype

int main()

{

// Call the showStatic function five times.

for (int count = 0; count < 5; count++)

showStatic();

return 0;

}

// Definition of function showStatic. statNum is a static local variable. Its value

// is displayed and then incremented just before the function returns.

void showStatic()

{

static int statNum;

cout << "statNum is " << statNum << endl;

statNum++;

}

// This program uses a reference variable as a function parameter.

#include <iostream>

using namespace std;

// Function prototype. The parameter is a reference variable.

void doubleNum(int &);

void addNum(int &);

void subNum(int &);

void multNum(int &);

void divNum(int &);

int main()

{

int value = 4;

cout << "In main, value is " << value << endl;

cout << "Now calling doubleNum..." << endl;

doubleNum(value);

cout << "Now back in main. value is " << value << endl;

addNum(value);

cout << "The added value is " << value << endl;

subNum(value);

cout << "The subtracted value is " << value << endl;

multNum(value);

cout << "The multiplied value is " << value << endl;

divNum(value);

cout << "The divided value is " << value << endl;

return 0;

}

// Definition of doubleNum. The parameter refVar is a reference variable. The

// value in refVar is doubled.

void doubleNum(int &refVar)

{

refVar \*= 2;

}

void addNum(int &refVar)

{

refVar += 2;

}

void subNum(int &refVar)

{

refVar -= 2;

}

void multNum(int &refVar)

{

refVar \*= 2;

}

void divNum(int &refVar)

{

refVar /= 2;

}\*/

// This program will write a function and return the temperature converted to

// Celsius.

#include <iostream>

#include <iomanip>

using namespace std;

double celsius(int fheight); // Function prototype

int main()

{

double temp = 0; // Celsius value

cout << "Fahrenheit\t\tCelsius" << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" << endl;

for (int fheight = 0; fheight <= 20; fheight++)

{

temp = celsius(fheight);

cout << fixed << fheight << "\t\t\t" << temp << "\n" << setprecision(2);

}

return 0;

}

double celsius(int fheight)

{

return ((0.556) \* (fheight - 32));

}