/\*

These are function declarations. These must be declared if the function definition comes after main(),

and if the functions are used in main();

This is because the compiler needs to know that these functions do exist, and that it needs to find

their definitions somewhere in the file.

\*/

#include <stdio.h>

void add(float a, float b); // You include the return type, the function name, and its parameters.

void subtract(float a, float b);

void divide(double a, double b);

void multiply(double a, double b);

int main()

{

printf("Please choose what operation\n"); // the /n will write a new line after being printed to the console

printf("1. Addition\n");

printf("2. Subtraction\n");

printf("3. Division\n");

printf("4. Multiplication\n");

printf("Type a number: ");

int choice;

scanf("%d", & choice);

// The %d means that this will accpet and integer input

if (choice == 1) // Addition

{

printf("Choose first number: ");

float x;

scanf("%f", & x); // This function will store the inputted value to x.

printf("Choose second number: ");

float y;

scanf("%f", & y);

add(x, y); // accepts the arguements x and y then adds them and prints out the result.

} else if (choice == 2) // Subtraction

{

printf("Choose first number: ");

float x;

scanf("%f", & x);

printf("Choose second number: ");

float y;

scanf("%f", & y);

subtract(x, y);

} else if (choice == 3) // Division

{

printf("Choose first number: ");

double x;

scanf("%lf", & x);

printf("Choose second number: ");

double y;

scanf("%lf", & y);

divide(x, y);

} else if (choice == 4) // Multiplication

{

printf("Choose first number: ");

double x;

scanf("%lf", & x);

printf("Choose second number: ");

double y;

scanf("%lf", & y);

multiply(x, y);

} else // If the user input wasn't an integer, this will be called

{

printf("That isn't recognized");

}

return 0;

}

/\*These are the actual function defintitions. Since the actual code of the functions

were defined here (after main()) we declared them earlier.\*/

// The return type for these functions are "void" because we aren't returning a value,

// just printing the calculated value. The printing is done by the actual function as well.

void add(float a, float b) // Addition function

{

float c = a + b;

printf("The sum is: %f", c);

// The %f is used to print floating point numbers

}

void subtract(float a, float b) // Subtraction function

{

float c = a - b;

printf("The difference is: %f", c);

}

// Why doubles instead of float? Doubles are more percise, but deciding between

// either of those is a decision thats left up to the programmer and situation.

// There is no right "type" its up to you.

void divide(double a, double b) // Division function

{

double c = a / b;

printf("The quotient is: %lf", c);

// The %lf is used for doubles

}

void multiply(double a, double b) // Multiply function

{

double c = a \* b;

printf("The product is: %lf", c);

}