TITLE

Debugging Code

LAB # 04

SECTION # 08

FULL NAME

Adam Jennissen

SUBMISSION DATE:

09/29/2022

DATE

09/27/2022

Problem

The purpose of this lab was to learn about the C compiler messages, become familiar with various types of compiler errors, and to learn coding practices to help avoid unintentional errors. The lab had us debug a total of 11 programs, 5 had compiler errors, 5 had logic errors, and the last one had a combination of both.

Analysis

In order to successfully do this lab, I had to figure out what the program was supposed to do and then figure out why it was not doing that. The compiler errors were straight forward, the compiler mostly explains what is wrong. The logic errors require a bit of thought about what it is supposed to do and then what it is currently doing.

Design

When doing the lab, I had to edit lines to their correct form and figure out why they did not work.

(1) 1_1. On line 33 I added in the second “ mark in order to tell the printf function where it should end. On line 35 I added a ; to the end of the line, so that the compiler knows it is the end of that line. On line 43 I added a { to tell the compiler what is part of the else function. On line 46 I added the n into the pritf to make it printf which is the correct function.

(2) 1_2. On line 35 I added double acceleration; I added this so that the acceleration variable was declared.

(3) 1_3. On line 13 I added #include <stdio.h> so that the standard input and output library can be used. On line 14 I added #include <stdlib.h> so that the rand function can be used. On line 20 I added void print_face(int selection); so that the main function can call the print_face function.

(4) 1_4. I had to change the names of 3 of the variables throughout the entire program. I changed speed_of_light! to speed_of_light because variable names cannot include !. I changed wave-length to wave_length, because variable names cannot include -. I changed ~length_in_meters to length_in_meters, because ~ cannot be used in variable names. I changed 0energy to energy, because variable name cannot start with a number. On line 35 I had to move const to before double, because that is the proper way to declare a constant variable.

(5) 1_5. On line 19 I removed the prototype for the main function, because it is not needed. On line 44-47 I removed the function, because you can only have one main function.

(6) 2_1. On line 35 I removed the second =, because you use a single = to assign a value. On line 40 and 46 I added an = because you use == to check if 2 values are equal.

(7) 2_2. On lines 57, 62, 67, 72, 77, 82, 87, and 92, I removed the (double), because you do not need to keep the variable as an int in order to figure out if it has a remainder or not.

(8) 2_3. On line 38 I changed both of the %lf to %d, so that it took the correct type of input.

(9) 2_4. On line 38 I changed the in to double, so that it used the correct type of variable.

(10) 2_5. On line 56 I added a second & and | because you need to use 2, because that is the proper syntax for or and and. On lines 81, 101, and 124, I added an else statement, so that it only runs if the if statement is false.

(11) 3. On line 13 I added #include <stdlib.h> so that I can use the rand function. On line 16 I added an * so that the comment is done properly. On line 23 I added void run_game(int computer_number); so that the function can be called in the main function. On line 34 I added a / to end the comment. On line 48 I added an e so that played was spelled correctly. On line 72 I added an &, because it is required when taking input for anything other than strings. On line 93 I added a +1 so that the rand function returns a value within 1-100. On line 105, I declared the variable correct with a type of int. On line 110 I changed the input variable type specifier from %c to %d. On line 119 I added an = sign because you use == when checking if 2 things are equal. On line 127 I removed the ; because you do not use one at the end of an else if statement. On line 133, I changed it to an else if statement so that I was able to check certain parameters.

Testing

When going through the lab I made sure to compile and run after changing one or two lines, to see if what I changed had fixed the problem. I also made sure that when I was running the problem I tested multiple inputs, both within the range and outside of the input range, to make sure the program handled them correctly. I also had to check after every change I made on the last 6 problems, because the errors in those ones were logic based, I had to be extra sure that they were running correctly.

Comments

I learned that using the -wall flag when compiling will tell you about things that could give unintended results. I got a much better understanding of what kind of error the compiler will notice, and what kind of errors will run, but give you improper results.

Screen Shots

1.

```c
#include <stdio.h>

/*-----------------------------------------------------------------
-                                   Notes
-----------------------------------------------------------------
// Compile with gcc lab04-1_1.c -o lab04-1_1
// Run with ./lab04-1_1
/* This program outputs if a integer will divide into another int

/*-----------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------
int main(int argc, char *argv[])
{
    int i, j;

    // printf("Enter an integer: ")
    printf("Enter an integer: ");//added semicoln
    scanf("%d", &i);

    // printf("Enter another integer: );
    printf("Enter another integer: ");//added double quotation ma
    // scanf("%d", &j)
    scanf("%d", &j);//added semicoln


    if (j % i == 0)
    {
        printf("%d divides %d\n", i, j);

    } else
    {//added curly bracket

        // pritf("%d does not divide %d\n", i, j);
        printf("%d does not divide %d\n", i, j);//added the n in
        printf("%d %% %d is %d\n", j, i, (j % i));
    }

    return 0;
}
```

```
/cygdrive/u/fall2022/se185/lab04
      |         ^~~~~~
      |         printf
/usr/lib/gcc/x86_64-pc-cygwin/11/../../../../x86_64-pc-cygwin/bin/ld: /tmp/cc6A
 `pritf'
collect2: error: ld returned 1 exit status

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ gcc lab04-1_1.c -o test

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Enter an integer: 3
Enter another integer: 6
3 divides 6

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Enter an integer: 4
Enter another integer: 7
4 does not divide 7
7 % 4 is 3

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$
```

2.

```c
// Run with ./lab04-1_2
/* This program takes two inputs, acceleration and mass,
 * and outputs the force = mass * acceleration */

/*-----------------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    double mass;
    double acceleration;//declared acceleration in this scope

    printf("Enter an acceleration in m/s^2: ");
    scanf("%lf", &acceleration);

    printf("Enter the mass of the object in kg: ");
    scanf("%lf", &mass);

    printf("\nYou entered %lf m/s^2.\n", acceleration);
    printf("You entered %lf kg.\n\n", mass);

    force(mass, acceleration);

    return 0;
}

/**
 * Given mass and acceleration, calculates the force exerted.
 *
 * @param mass - The given mass of an object in kilograms.
 * @param acceleration - The acceleration of an object in m/s^2.
 */
void force(double mass, double acceleration)
{
    printf("The force is approximately %.2lf Newtons.\n", mass * accelerati
}
```

```
adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ gcc lab04-1_2.c -o test

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Enter an acceleration in m/s^2: 6
Enter the mass of the object in kg: 3000

You entered 6.000000 m/s^2.
You entered 3000.000000 kg.

The force is approximately 18000.00 Newtons.
```

3.

lab04-1_3.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-1_3.c

```c
/*-----------------------------------------------------------------------
-                                Includes
-----------------------------------------------------------------------
#include <time.h>
#include <stdio.h>//added the standard input and output library
#include <stdlib.h>//added the standard library in order to use rand functi

/*-----------------------------------------------------------------------
-                                Prototypes
-----------------------------------------------------------------------
void hoo();
void print_face(int selection);//added prototype for the function

/*-----------------------------------------------------------------------
-                                    Notes
-----------------------------------------------------------------------
/* This is a simple program that takes a user inputs
 * and prints out a message based on that input */
// Compile with gcc lab04-1_3.c -o lab04-1_3
// Run with ./lab04-1_3

/*-----------------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    srand(time(NULL));

    int selection = 0;

    printf("Enter 1 for happy, 2 for sad, 3 for neutral, any other integer
    scanf("%d", &selection);

    if (selection < 1 || selection > 3)
    {
        selection = rand() % 4;
    }

    print_face(selection);

    return 0;
```

Ln 20, Col 59 Spaces: 4 UTF-8 LF C

/cygdrive/u/fall2022/se185/lab04

```
adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Enter 1 for happy, 2 for sad, 3 for neutral, any other integer for random: 3
Meh :\

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Enter 1 for happy, 2 for sad, 3 for neutral, any other integer for random: 2
:(

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Enter 1 for happy, 2 for sad, 3 for neutral, any other integer for random: 7
Have a nice day! :)

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$
```

4.

lab04-1_4.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-1_4.c

```c
/* This program calculates the energy of one photon
 * of user-inputted wave-length of light */

/*-----------------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    // double speed_of_light!;
    double speed_of_light;//removed !
    // double wave-length;
    double wave_length;//replaced - with _
    // double ~length_in_meters;
    double length_in_meters;//removed ~
    // double plank const;
    const double plank = 6.62606957 * pow(10, -34);//moved const to before
    // double 0energy;
    double energy;//removed 0


    // plank const = 6.62606957 * pow(10, -34); // Planck's constant

    // speed_of_light! = 2.99792458 * pow(10, 8); // Constant for the speed
    speed_of_light = 2.99792458 * pow(10, 8); // Constant for the speed of

    // wave-length = 0;
    // ~length_in_meters = 0;
    wave_length = 0;//replaced - with _
    length_in_meters = 0;//removed ~
    // 0energy = 0;
    energy = 0;//removed 0


    printf("Welcome! This program will give the energy, in Joules,\n");
    printf("of 1 photon with a certain wave-length.\n");
    printf("Please input a wave-length of light in nano-meters.\n");
    printf("Please do not enter a negative, or zero, wave-length.\n");

    // scanf("%lf", &wave-length);
    scanf("%lf", &wave_length);//replaced - with _
```

Ln 35, Col 100 Spaces: 4 UTF-8 LF C

/cygdrive/u/fall2022/se185/lab04

```
   35 |     const double plank;//moved const to before plank
      |                  ^~~~~

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ gcc lab04-1_4.c -o test

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Welcome! This program will give the energy, in Joules,
of 1 photon with a certain wave-length.
Please input a wave-length of light in nano-meters.
Please do not enter a negative, or zero, wave-length.
500
A photon with a wave-length of 0500.000 nano-meters, carries
approximately 0000.000000000000000003972891 joules of energy.
adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$
```

5.

lab04-1_5.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-1_5.c

```c
-                                   Notes
-----------------------------------------------------------------------
// Compile with gcc lab04-1_5.c -o lab04-1_5
// Run with ./lab04-1_5
/* This program calculates the sum of 1 to x, where x is a user input */

/*---------------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    int input;

    printf("Please input a number from to sum up to: ");

    scanf("%d", &input);

    printf("The sum of 1 to %d is %d\n", input, sum_function(input));

    return 0;
}

// int main(int argc, char *argv[])//not needed
// {
//     printf("Sum is 32!\n");
// }

/**
 * Calculates the sum of 1 to number of a given number.
 *
 * @param number - The number that determines what the sum will stop adding
 * @return - The sum of 1 to the given number.
 */
int sum_function(int number)
{
    return (number * (number + 1)) / 2;
}
```

Ln 44, Col 48 Spaces: 4 UTF-8 LF C

/cygdrive/u/fall2022/se185/lab04

```
lab04-1_5.c:44:5: error: redefinition of 'main'
   44 | int main(int argc, char *argv[])
      |     ^~~~
lab04-1_5.c:31:5: note: previous definition of 'main' with type 'int(int,  char **)'
   31 | int main(int argc, char *argv[])
      |     ^~~~

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ gcc lab04-1_5.c -o test

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input a number from to sum up to: 4
The sum of 1 to 4 is 10

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$
```

6.

lab04-2_1.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-2_1.c

```c
// Run with ./lab04-2_1
/* This program accepts a user input and determines
 * if the integer is an odd or an even number */

/*-----------------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    // int input == 0;
    int input = 0;//removed = so it is assigning the value

        printf("Please input an integer: ");
        scanf("%d", &input);
    // if (is_odd(input) = 1)
    if (is_odd(input) == 1)//added = to check if its equal
    {
        printf("%d is an odd number!\n", input);
    }

    // if (is_even(input) = 1)
    if (is_even(input) == 1)//added = to check if its equal
    {
        printf("%d is an even number!\n", input);
    }

    return 0;
}

/**
 * Determines whether the given number is even.
 *
 * @param number - The number in question of even status.
 * @return - True if the given number was even.
 */
int is_even(int number)
{
    return !(number % 2);
}

/**
```

/cygdrive/u/fall2022/se185/lab04

```
adamjenn@C01318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input an integer: 3
3 is an odd number!

adamjenn@C01318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input an integer: 6
6 is an even number!

adamjenn@C01318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input an integer: 0
0 is an even number!

adamjenn@C01318-24 /cygdrive/u/fall2022/se185/lab04
$
```

7.

lab04-2_2.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-2_2.c

```c
        printf("8 digits\n");
    }
    // else if ((double) number / 1000000 != 0)
    else if ( number / 1000000 != 0)//removed double
    {
        printf("7 digits\n");
    }
    // else if ((double) number / 100000 != 0)
    else if ( number / 100000 != 0)//removed double
    {
        printf("6 digits\n");
    }
    // else if ((double) number / 10000 != 0)
    else if ( number / 10000 != 0)//removed double
    {
        printf("5 digits\n");
    }
    // else if ((double) number / 1000 != 0)
    else if ( number / 1000 != 0)//removed double
    {
        printf("4 digits\n");
    }
    // else if ((double) number / 100 != 0)
    else if ( number / 100 != 0)//removed double
    {
        printf("3 digits\n");
    }
    // else if ((double) number / 10 != 0)
    else if ( number / 10 != 0)//removed double
    {
        printf("2 digits\n");
    }
    // else if ((double) number / 1 != 0)
    else if (number / 1 != 0)//removed double
    {
        printf("1 digit\n");
    }
}
```

Ln 92, Col 12 Spaces: 4 UTF-8 LF C

/cygdrive/u/fall2022/se185/lab04

```
adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input an integer from 1 up to 10000000: 3032918
7 digits

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input an integer from 1 up to 10000000: 324
3 digits

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input an integer from 1 up to 10000000: 34556
5 digits

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$
```

8.

lab04-2_3.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-2_3.c

```c
 * swaps their values using two different methods */
// Compile with gcc lab04-2_3.c -o lab04-2_3
// Run with ./lab04-2_3

/*-----------------------------------------------------------------------
-                                Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    int first = 0, second = 0;
    printf("Please input two integers separated by a space: ");

    // scanf("%lf %lf", &first, &second);
    scanf("%d %d", &first, &second);//changed input type

    printf("\n");
    variable_swap(first, second);

    printf("\n");
    math_swap(first, second);

    return 0;
}

/**
 * Swaps the values of two integers using a temp variable.
 *
 * @param i - The first value to be swapped.
 * @param j - The second value to be swapped.
 */
void variable_swap(int i, int j)
{
    printf("Now doing a swap using an extra variable: \n");
    printf("Before Swap: First: %d, Second: %d\n", i, j);

    int temp = i;
    i = j;
    j = temp;

    printf("After Swap: First: %d, Second: %d\n", i, j);
}
```

/cygdrive/u/fall2022/se185/lab04

```
adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ gcc lab04-2_3.c -o test

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please input two integers separated by a space: 3 4

Now doing a swap using an extra variable:
Before Swap: First: 3, Second: 4
After Swap: First: 4, Second: 3

Now doing a swap using addition and subtraction:
Before Swap: First: 3, Second: 4
After Swap: First: 4, Second: 3

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$
```

9.

lab04-2_4.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-2_4.c

```c

/*-----------------------------------------------------------------------
-                                Notes
-----------------------------------------------------------------------
// Compile with gcc lab04-2_4.c -o lab04-2_4
// Run with ./lab04-2_4
/* This program calculates values of resistances,
 * voltages, or current using Ohm's Law */

/*-----------------------------------------------------------------------
-                              Implementation
-----------------------------------------------------------------------
int main(int argc, char *argv[])
{
    int selection = 0;
    // int v, i, r;
    double v, i, r;//changed type to double

    printf("selection:\n1 for voltage\n2 for resistance\n3 for current\n");

    scanf("%d", &selection);

    if (selection > 3 || selection < 1)
    {
        printf("Invalid number\n");
        return -1;
    }

    printf("Enter floating point numbers for input...\n");
    if (selection == 1)
    {
        printf("Please enter a resistance value: ");
        scanf("%lf", &r);

        printf("Please enter a current value: ");
```

/cygdrive/u/fall2022/se185/lab04

```
3 for current
1
Enter floating point numbers for input...
Please enter a resistance value: 3.764
Please enter a current value: 400
Your voltage is: 1505.600000 Volts

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
selection:
1 for voltage
2 for resistance
3 for current
3
Enter floating point numbers for input...
Please enter a resistance value: 234
Please enter a voltage value: 4
Your current is: 0.017094 Amps

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
selection:
1 for voltage
2 for resistance
3 for current
5
Invalid number
```

10.

lab04-2_5.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-2_5.c

```c
    if (number > 10000 | number < -10000)
    {
        printf("Number is out of range!\n");
        return -1;
    }

    // if ((is_positive(number) & !is_negative(number)) | is_zero(number))
    if ((is_positive(number) && !is_negative(number)) || is_zero(number))//
    {
        printf("%d is a whole number.\n", number);
    }
    else
    {
        printf("%d is non-whole number.\n", number);
    }

    return 0;
}

/**
 * Determines if the given number is positive.
 *
 * @param number - The number in question of whether it is positive or not.
 * @return - Whether the given number is positive.
 */
int is_positive(int number)
{
    if (number > 0)
    {
        printf("%d is positive and ", number);
        return 1;
    }
else//added else
{
    printf("%d is non-positive and ", number);
    return 0;
}
}

/**
 * Determines if the given number is negative
```

```
adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please type a number between -10000 and 10000: -500
-500 is non-positive and -500 is non-zero and -500 is non-whole number.

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please type a number between -10000 and 10000: 35
35 is positive and 35 is non-negative and 35 is a whole number.

adamjenn@CO1318-24 /cygdrive/u/fall2022/se185/lab04
$ ./test
Please type a number between -10000 and 10000: -43
-43 is non-positive and -43 is non-zero and -43 is non-whole number.
```

11.

lab04-3.c - u: - Visual Studio Code

fall2022 > se185 > lab04 > lab04-3.c

```c
            printf("Do you want to play a game?
                "Enter 'y' to play, anything else not to play. :(\n  -> ");
            // scanf(" %c", yes_or_no);
                scanf(" %c", &yes_or_no);//added &

        } else
        {
            scanf(" %c", &yes_or_no);
        }

        printf("%c", yes_or_no);

        return yes_or_no;
    }

    /**
     * Generates a random number between 1 to 100, inclusive.
     *
     * @return - A number between 1 and 100, inclusive.
     */
    int select_random_number()
    {
        srand(time(NULL));
        // return rand() % 100;
        return ((rand() % 100)+1);//changed so the rand is within 1-100

    }

    /**
     * Starts the Guessing Game for you to play!
     *
     * @param computer_number - The randomly generated number to be used for th
     */
    void run_game(int computer_number)
    {
        int number = 0;
        int correct = 0;//declared variable

        printf("\n\nYou are guessing a number. The options are 1 through 100.\n
        printf("What is your guess on what number I will select?\n  -> ");
        // scanf("%c", &number);
        scanf("%d", &number);//changed the type specifier
```

/cygdrive/u/fall2022/se185/lab04

```
You guessed too high. Enter another guess.
 -> 57

You guessed too high. Enter another guess.
 -> 42

You guessed too high. Enter another guess.
 -> 32

You guessed too high. Enter another guess.
 -> 15

You guessed too low. Enter another guess.
  -> 25

The number was 25!
```