



NEW YORK CITY COLLEGE OF TECHNOLOGY

THE CITY UNIVERSITY OF NEW YORK

300 JAY STREET, BROOKLYN, NY 11201-1909

Department of Computer Engineering Technology

CET 3510 - OL25 - Microcomputer Systems Technology lab

LABORATORY REPORT No_02

Lab_No_02 Data Format and Date Conversion

Professor: Dr. Rubén Velásquez, Ph.D.

Spring 2021

Student's Name: Ashahi Shafin

Student's ID: 23607352

Preparation date:

February 17, 2021

Due date:

February 24, 2021

1. Table of content

- 1. Table of content*
- 2. Objective*
- 3. Laboratory tools to perform the task*
- 4. Source code*

Laboratory No_02 Example 2.2

- 5. Source code Line Description*

Laboratory No_02 Example 2.2

- 6. Explaining the output results*

Laboratory No_02 Example 2.2

- 7. Program Flow Chart*

- 8. Table*

- 9. Comments*

- 10. Conclusion*

2. OBJECTIVE

Laboratory No 2. The basic storage unit for all data in x86 computer is a byte. Find the integer storage size in byte(s) and the maximum and minimum values for different data types. Understand data formats of signs numbers and unsigned numbers practice number conversion among decimal, hexadecimal, and binary numbers.

3. LABORATORY TOOLS TO PERFORM THE TASK

Computer NZXT

Microsoft visual studio 2019 Software

Microsoft Word 2019 Software

Microsoft Notepad Software

4. SOURCE CODE

Laboratory No_01 Example 2.2

Source Code

```
#include "stdio.h"
#include<iostream>
#include<bitset>

using namespace std;

int main(void)
{
    printf(" Lab_No_02_Data_Formats_and_Data_Conversion\n");
    printf(" Module: C++ programming \n");
    printf(" Ashahi Shafin, ID#23607352\n");
    printf(" CET3510-OL25\n");
    printf(" Presentation date: February 17, 2021\n");
    printf(" Due date: February 24, 2021\n");
    printf(" Example 2.2 dataFormatTemp.cpp\n");
    printf(" file name: 2.2 dataFormatTemp.cpp\n");
    printf("-----\n");

    char c1;
    unsigned char uc1;
    short int i1;
    signed char sc1;
    unsigned short us1;
    unsigned int ui1;
```

```

//Data type of char
cout << "The minimun value of a char" << CHAR_MIN << endl;
cout << "The maximun value of a char" << CHAR_MAX << endl;
cout << "The stoeage bytes of a char is" << sizeof(c1) << endl;

cout << "Input a hexideceecimal munchber in the data type, for example 8a" << endl;
scanf_s("%x", &c1)
// bitset display the number of bits in 8*sizeof(c1);
bitset<8 * sizeof(c1)> charBits(c1);
cout << "The converted binary valueis " << charBits << endl;
printf("The converted decimal value is %i \n", c1);
printf("-----\n");

//Data type of signed char
cout << "The minimun value of a signed char" << UCHAR_MAX << endl;
cout << "The stoeage bytes of a sigened char is" << sizeof(uc1) << endl;

cout << "Input a hexideceecimal munchber in the data type, for example 8a" << endl;
scanf_s("%x", &uc1)
// bitset display the number of bits in 8*sizeof(uc1)
bitset<8 * sizeof(uc1)> ucharBits(uc1);
cout << "The converted binary valueis " << ucharBits << endl;
printf("The converted decimal value is %i \n", uc1);
printf("-----\n");

//Data type of signed short int
cout << "The minimun value of a short int" << SHRT_MIN << endl;
cout << "The maximun value of a short int" << SHRT_MAX << endl;
cout << "The stoeage bytes of a short int is" << sizeof(i1) << endl;

cout << "Input a hexideceecimal munchber in the data type, for example 8a" << endl;
scanf_s("%x", &i1)
// bitset display the number of bits in 8*sizeof(i1)
bitset<8 * sizeof(i1)> shortBits(i1);
cout << "The converted binary valueis " << shortBits << endl;
printf("The converted decimal value is %i \n", i1);
printf("-----\n");

//Data type of signed int
cout << "The minimun value of a signed char" << SCHAR_MIN << endl;
cout << "The maximun value of a signed char" << SCHAR_MAX << endl;
cout << "The stoeage bytes of a signed char is" << sizeof(sc1) << endl;

cout << "Input a hexideceecimal munchber in the data type, for example 8a" << endl;
scanf_s("%x", &sc1)
// bitset display the number of bits in 8*sizeof(sc1)
bitset<8 * sizeof(sc1)> scharBits(sc1);
cout << "The converted binary valueis " << scharBits << endl;
printf("The converted decimal value is %i \n", sc1);
printf("-----\n");

//Data type of unsigned short int
cout << "The minimun value of a signed char" << USHRT_MAX << endl;
cout << "The stoeage bytes of a sigened char is" << sizeof(us1) << endl;

cout << "Input a hexideceecimal munchber in the data type, for example 8a" << endl;
scanf_s("%x", &us1)
// bitset display the number of bits in 8*sizeof(us1)
bitset<8 * sizeof(us1)> ushortBits(us1);

```

```

cout << "The converted binary value is " << ushortBits << endl;
printf("The converted decimal value is %i \n", us1);
printf("-----\n");

// Data type of signed unsigned int
cout << "The minimum value of a unsigned int" << INT_MIN << endl;
cout << "The maximum value of a unsigned int" << INT_MAX << endl;
cout << "The storage bytes of a unsigned int is" << sizeof(ui1) << endl;

cout << "Input a hexadecimal number in the data type, for example 8a" << endl;
scanf_s("%x", &ui1)
    // bitset display the number of bits in 8*sizeof(ui1)
    bitset<8 * sizeof(ui1)> intBits(ui1);
cout << "The converted binary value is " << intBits << endl;
printf("The converted decimal value is %i \n", ui1);
printf("-----\n");

system("pause");
exit(0);
return 0;
}

```

5. SOURCE CODE LINE DESCRIPTION

Laboratory No_02 Example 2.2

<i>Line</i>	<i>Source Code Description</i>
01	#include "stdio.h"
	The stdio.h (standard library header) is a file with ".h" extension that contains the prototypes of standard input-output functions used in C.
02	#include<iostream>
	iostream provides basic input and output services for C++ programs. iostream uses the objects cin , cout , cerr , and clog for sending data to and from the standard streams input, output, error (unbuffered), and log (buffered) respectively.
03	#include<set>
	Set is a container in C++ Standard Template Library for dealing with data at the bit level.
04	using namespace std;
	So when we run a program to print something, “using namespace std” says if you find something that is not declared in the current scope go and check std. using namespace std; are used. It is because computer needs to know the code for the cout, cin functionalities and it needs to know

which **namespace** they are defined.

05 int main(void)
 {
 MAIN PROGRAM
 }

In C and C++ **int main(void)** means that the function takes NO arguments. ... **Int main(void)** is used in C to restrict the function to take any arguments, if you do not put **void** in those brackets, the function will take ANY number of arguments you supply at call.

06 printf("*****\n")

Output console to indicate separation marked with asterisks within parentheses.

07 char c1;
 unsigned char uc1;
 short int i1;
 signed char sc1;
 unsigned short us1;
 unsigned int ui1;

A **variable's type** determines the values that the **variable** can have and the operations that can be performed on it.

08 cout << "The minimum value of a char" << CHAR_MIN << endl;
 cout << "The maximum value of a char" << CHAR_MAX << endl;
 cout << "The storage bytes of a char is" << sizeof(c1) << endl;
 cout << "Input a hexadecimal number in the data type, for example 8a" << endl;

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

09 scanf_s("%x", &c1)

The **scanf_s()** function reads data from the standard input stream stdin and writes the data into the location given by argument. Each argument must be a pointer to a variable of a type that corresponds to a type specifier in format. If copying takes place between strings that overlap, the behavior is undefined.

10 bitset<8 * sizeof(c1)> charBits(c1);

Bitset is a container in C++ Standard Template Library for dealing with data at the bit level.

11 cout << "The converted binary value is " << charBits << endl;

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

12 printf("The converted decimal value is %i \n", c1);

Output console display information within parenthesis and what in c1.

13 cout << "The minimum value of a signed char" << UCHAR_MAX << endl;
cout << "The storage bytes of a signed char is" << sizeof(uc1) << endl;
cout << "Input a hexadecimal number in the data type, for example 8a" << endl;

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

14 scanf_s("%x", &uc1)

The **scanf_s()** function reads data from the standard input stream stdin and writes the data into the location given by argument. Each argument must be a pointer to a variable of a type that corresponds to a type specifier in format. If copying takes place between strings that overlap, the behavior is undefined.

15 bitset<8 * sizeof(uc1)> ucharBits(uc1);

Bitset is a container in C++ Standard Template Library for dealing with data at the bit level.

16 cout << "The converted binary value is " << ucharBits << endl;

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

17 printf("The converted decimal value is %i \n", uc1);

Output console display information within parenthesis in uc1.

18 cout << "The minimum value of a short int" << SHRT_MIN << endl;
cout << "The maximum value of a short int" << SHRT_MAX << endl;
cout << "The storage bytes of a short int is" << sizeof(i1) << endl;
cout << "Input a hexadecimal number in the data type, for example 8a" << endl;

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

19 scanf_s("%x", &i1)

The **scanf_s()** function reads data from the standard input stream stdin and writes the data into the location given by argument. Each argument must be a pointer to a variable of a type that corresponds to a type specifier in format. If copying takes place between strings that overlap, the behavior is undefined.

20 bitset<8 * sizeof(i1)> shortBits(i1);

Bitset is a container in C++ Standard Template Library for dealing with data at the bit level.

21 cout << "The converted binary value is " << shortBits << endl;

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

22 printf("The converted decimal value is %i \n", i1);

Output console display information within parenthesis in i1.

23

```
cout << "The minimum value of a signed char" << SCHAR_MIN << endl;
cout << "The maximum value of a signed char" << SCHAR_MAX << endl;
cout << "The storage bytes of a signed char is" << sizeof(sc1) << endl;
cout << "Input a hexadecimal number in the data type, for example 8a" << endl;
```

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

24

```
scanf_s("%x", &sc1)
```

The **scanf_s()** function reads data from the standard input stream stdin and writes the data into the location given by argument. Each argument must be a pointer to a variable of a type that corresponds to a type specifier in format. If copying takes place between strings that overlap, the behavior is undefined.

25

```
bitset<8 * sizeof(sc1)> scharBits(sc1);
```

Bitset is a container in C++ Standard Template Library for dealing with data at the bit level.

27

```
cout << "The converted binary value is " << scharBits << endl;
```

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

28

```
printf("The converted decimal value is %i \n", sc1);
```

Output console display information within parenthesis in sc1.

29

```
cout << "The minimum value of a signed char" << USHRT_MAX << endl;
cout << "The storage bytes of a signed char is" << sizeof(us1) << endl;
cout << "Input a hexadecimal number in the data type, for example 8a" << endl;
```

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

30

```
scanf_s("%x", &us1)
```

The **scanf_s()** function reads data from the standard input stream stdin and writes the data into the location given by argument. Each argument must be a pointer to a variable of a type that corresponds to a type specifier in format. If copying takes place between strings that overlap, the behavior is undefined.

31

```
bitset<8 * sizeof(us1)> ushortBits(us1);
```

Bitset is a container in C++ Standard Template Library for dealing with data at the bit level.

32

```
cout << "The converted binary value is " << ushortBits << endl;
```

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

33

```
printf("The converted decimal value is %i \n", us1);
```

Output console display information within parenthesis in uc1.

34

```
cout << "The minimun value of a unsigned int" << INT_MIN << endl;
cout << "The maximun value of a unsigned int" << INT_MAX << endl;
cout << "The stoeage bytes of a unsigned int is" << sizeof(ui1) << endl;
cout << "Input a hexideecimal munchber in the data type, for example 8a" << endl;
```

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

35

```
scanf_s("%x", &ui1)
```

The **scanf_s()** function reads data from the standard input stream stdin and writes the data into the location given by argument. Each argument must be a pointer to a variable of a type that corresponds to a type specifier in format. If copying takes place between strings that overlap, the behavior is undefined.

36

```
bitset<8 * sizeof(ui1)> intBits(ui1);
```

Bitset is a container in C++ Standard Template Library for dealing with data at the bit level.

37

```
cout << "The converted binary valueis " << intBits << endl;
```

The **cout** object in C++ is an object of class ostream. It is used to display the output to the standard output device i.e. monitor. It is associated with the standard C output stream stdout.

38

```
printf("The converted decimal value is %i \n", ui1);
```

Output console display information within parenthesis in ui1.

39

```
system("pause");
```

System("pause") runs the Windows command-line "**pause**" program and waits for that to terminate before it continues execution of the program , the console window stays open so you can read the output.

40

```
exit(0);
```

In the C Programming Language, the **exit function** calls all **functions** registered with atexit and terminates the program. File buffers are flushed, streams are closed, and temporary files are deleted.

41

```
return 0;
```

Terminates the execution of a function and **returns** control to the calling function (or to the operating system if you transfer control from the main function). Execution resumes in the calling function at the point immediately following the call.

6. EXPLAINING OUTPUT RESULTS

Laboratory No_02 Example 2.2

- i. This is the output console of example 2.2 for laboratory 2. It show the CHAR_MIN and MAX UCHAR_MAX SHRT_MIN and MAX SCHAR_MIN and MAX USHRT_MAX INT_MIN and MAX in 8 bit however this lab would be longer if we have added the 16 and 32 bit display and would take too long to explain. They would have also make the output very long.

```
LAB_No_02_Data_Format_and_Data_Conversion
```

```
Module: C++ Programming approach
```

```
Ashahi Shafin, ID# 23607352
```

```
CET3510-OL25
```

```
Submission date: February 24, 2021
```

```
Due date: February 24, 2021
```

```
Example 2.2 dataFormattemp.cpp
```

```
file name:LAB_2_DATA_FORMATS_AND_DATA_CONVERSION
```

```
-----
```

```
The minimum value of a signed char is -128
```

```
The Maximum value of a signed char is 127
```

```
The storage size in byte(s) of a signed char is 1
```

```
Input a hexadecimal number in the data type of signed char , for example 8a
```

```
8a
```

```
The converted binary value is 10001010
```

```
The converted decimal value is -118
```

```
-----
```

```
The Minimum value of a signed short int -2147483648
```

```
The Maximum value of a signed short int 2147483647
```

```
The storage size in byte(s) of a signed short int is 2
```

```
Input a hexadecimal number in the data type of signed short int, for example 8fff
```

```
8a
```

```
The converted binary value in 8 bits is 0000000010001010
```

```
The converted decimal value is 138
```

```
-----
```

```
The Minimum value of a signed int -2147483648
```

```
The Maximum value of a signed int 2147483647
```

```
The storage size in byte(s) of a signed int is 4
```

```
Input a hexadecimal number in the data type of signed int, for example 8fff
```

```
8a
```

```
The converted binary value in 8 bits is 0000000000000000000000000000000010001010
```

```
The converted decimal value is 138
```

```
-----
```

```
The Maximum value of a unsigned short int 65535
```

```
The storage size in byte(s) of a unsigned short int is 2
```

```
Input a hexadecimal number in the data type of unsigned short int, for example 8fff
```

```
8a
```

```
the converted binary value 8 in bits is 0000000010001010
```

```
The converted decimal value is 138
```

```
The converted decimal value is 138
```

```
-----  
The Maximum value of a unsigned int 4294967295
```

```
The storage size in byte(s) of a unsigned int is 4
```

```
Input a hexadecimal number in the data type of unsigned int, for example 8fff
```

```
8a
```

```
The storage size in byte(s) of a unsigned short int is 2
```

```
Input a hexadecimal number in the data type of unsigned short int, for example 8fff
```

```
8a
```

```
the converted binary value 8 in bits is 0000000010001010
```

```
The converted decimal value is 138
```

```
The Maximum value of a unsigned int 4294967295
```

```
The storage size in byte(s) of a unsigned int is 4
```

```
Input a hexadecimal number in the data type of unsigned int, for example 8fff
```

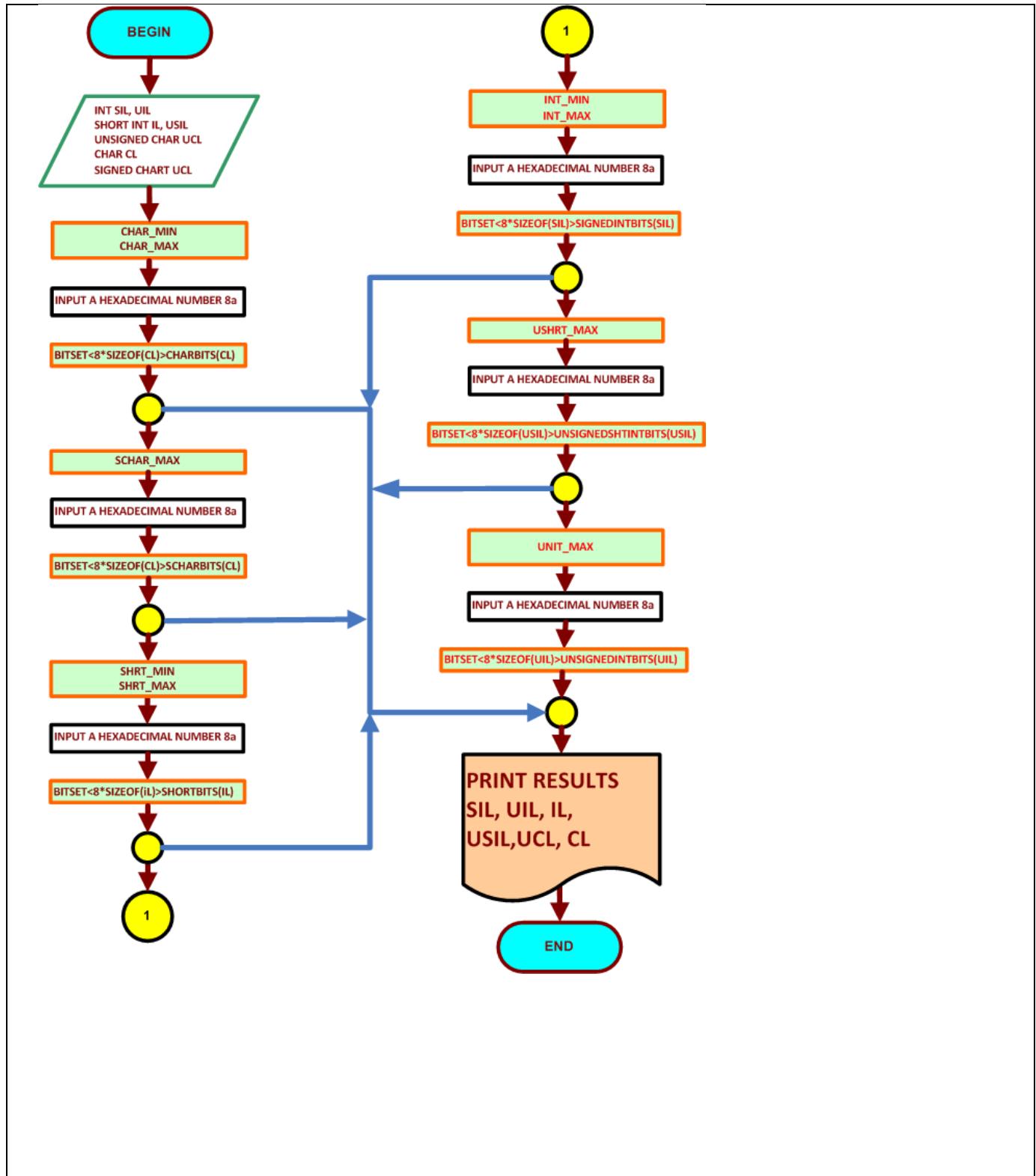
```
8a
```

```
The converted binary value in 8 bit is 000000000000000000000000000010001010
```

```
The converted decimal value is 138
```

```
-----  
Press any key to continue . . .
```

7. PROGRAM FLOW CHART:



8.

9. COMMENTS

Comments on Outputs shows how 8-bit conversion work while using variables and making work char short signed unsigned and int. with this we can see what are going to be max and min for these variables. And it show which bit rate we are using in our case we only used 8 bit

10. CONCLUSION

In this experiment, the module shows us the max and min for char short int unsigned char short char and unsigned short and after that it will ask us to convert 8a into binary and decimal. This was also used in a 8 bit method because if we did this 16 and 32 bits then the lab will be much longer and the output will be off the screen. The thing I had problem with this lab was coding it because I keep on getting errors and had to fix them in fact I had 14 errors and it was because I cout as count so I had hard time trying to see what I did wrong since count is real word I got confused.

A day in real life can be used with this code is a computer because they use many different type of bit like 8 16 32 64 and they use binary and decimals to communicate the files and max and min value can show how much they can hold in a drive. The computer also using this in order to transfer data and it can show you how bunch bit it taking to transfer and it will be very useful for a lot of people when they look into file transferring.