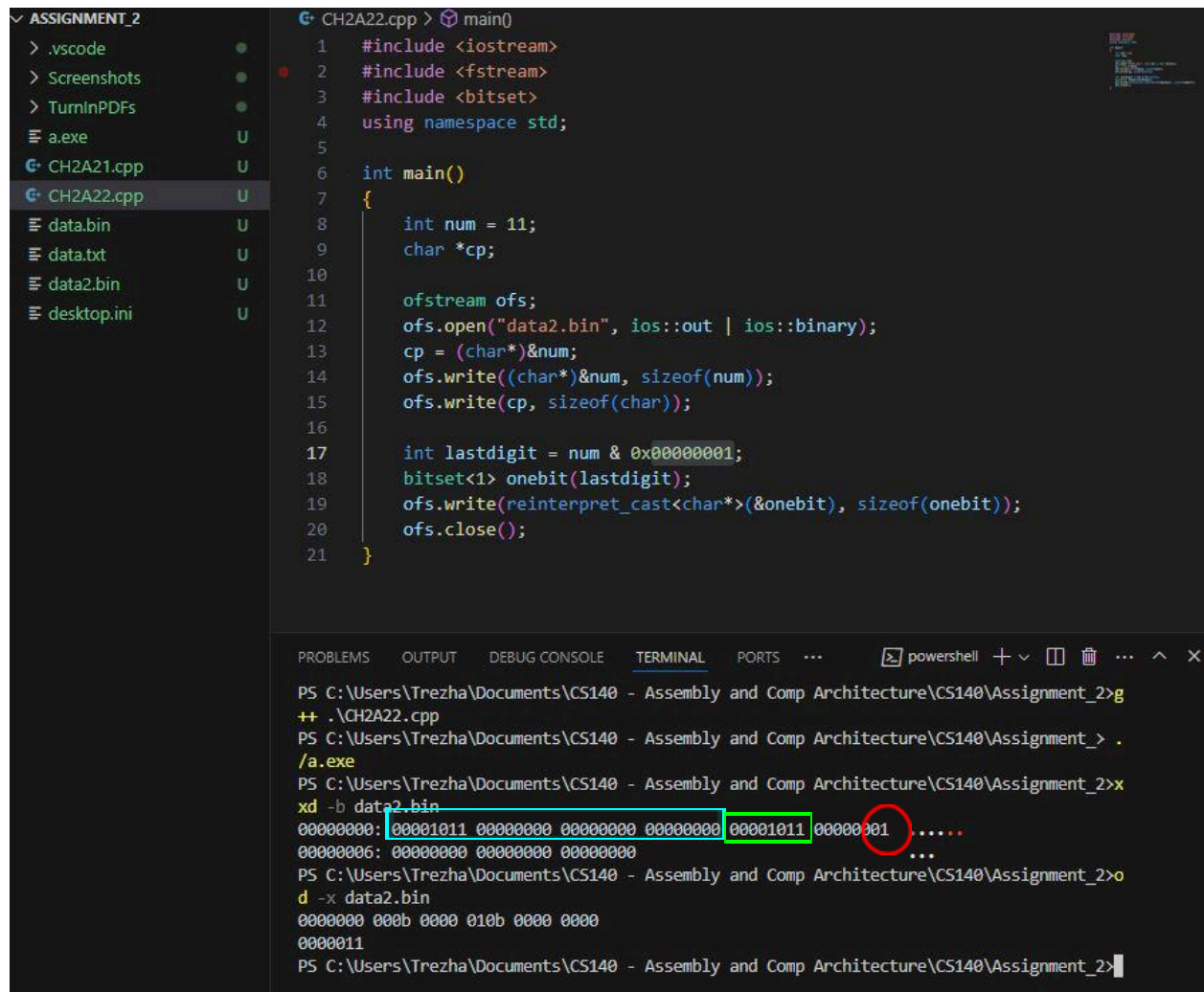


Assignment 2-2-2 The First Bit Location

First Bit Location



The screenshot shows a Visual Studio Code editor with a C++ file named CH2A22.cpp. The code defines a main function that writes the number 11 to a file named data2.bin in binary mode. It then isolates the least significant bit of the number 11 using a bitwise AND operation with 1. The terminal output shows the command prompt, the compilation of CH2A22.cpp, and the execution of the resulting a.exe. The hex dump output shows the binary data stored in data2.bin, with the least significant bit of the number 11 (0b00001011) highlighted in a green square and the final bit (1) circled in red.

```
CH2A22.cpp > main()
1  #include <iostream>
2  #include <fstream>
3  #include <bitset>
4  using namespace std;
5
6  int main()
7  {
8      int num = 11;
9      char *cp;
10
11     ofstream ofs;
12     ofs.open("data2.bin", ios::out | ios::binary);
13     cp = (char*)&num;
14     ofs.write((char*)&num, sizeof(num));
15     ofs.write(cp, sizeof(char));
16
17     int lastdigit = num & 0x00000001;
18     bitset<1> onebit(lastdigit);
19     ofs.write(reinterpret_cast<char*>(&onebit), sizeof(onebit));
20     ofs.close();
21 }
```

```
PS C:\Users\Trezha\Documents\CS140 - Assembly and Comp Architecture\CS140\Assignment_2>g
++ .\CH2A22.cpp
PS C:\Users\Trezha\Documents\CS140 - Assembly and Comp Architecture\CS140\Assignment_2> .
/a.exe
PS C:\Users\Trezha\Documents\CS140 - Assembly and Comp Architecture\CS140\Assignment_2>x
xd -b data2.bin
00000000: 00001011 00000000 00000000 00000000 00001011 00000001 .....
00000006: 00000000 00000000 00000000
PS C:\Users\Trezha\Documents\CS140 - Assembly and Comp Architecture\CS140\Assignment_2>o
d -x data2.bin
00000000 000b 0000 010b 0000 0000
00000111
PS C:\Users\Trezha\Documents\CS140 - Assembly and Comp Architecture\CS140\Assignment_2>
```

11 in Binary is 00001011

Note: **num** stores as **Little Endian**.

The first sequence stored in little Endian is 000b 0000

With 11(0b) being the LSB

So the **BLUE** square is what the binary file looks like before the code

`int lastdigit = num & 0x00000001;` executes

This line **isolates the least significant bit** (the “last digit” of num), for 11 in binary.

The **GREEN** square highlights this isolation 00001011(0b)

Finally going further, the **least significant bit in the least significant byte, is 1**.

Therefore “last digit” is 1(00000001), which is highlighted in the **RED** circle.
Which is also where the first bit is located.

This is how we got the final byte sequence of
00001011 00000000 00000000 00000000 00001011 00000001

Hexadecimal

11 in hex is 0b

000b 0000

Then after the line for lastdigit executes, it saves the LSB which is 1, into the second bit, so it's really just a representation of two things, 01 = LSB and 0b = 11

010b 0000

I assume the remaining 0s are padding, because we do see a 00000006 with 3 bytes. I tried my best to figure that part out but I have no idea. Hands up.