Lorenz Machine Bugs Located

# Bug 1

Line 12 in wheel.cpp is missing the “this->” pointer to refer the passing values to the current wheel class.

Text

Description automatically generated

## Fix:

Apply the correction to the left part of the value assignation.

# Bug 2

Line 25 in wheelGroup.cpp, inside the for loop the condition is not modifying the 5th wheel of the group at anytime.

Text

Description automatically generated

## Fix:

By removing the “-1” from the condition.

# Bug 3

Line 27 in wheelGroup.cpp, the direction of the bitwise operation is incorrect.

Text

Description automatically generated

## Fix:

Change the direction of the bitwise operation from >> to <<.

# Bug 5

Line 10 in wheel.cpp, building and running the program with the example “HELLO” brings back a result which has only 2 values. With that result the first approach is to check what is the application receiving and it is receiving 0 & 1 but the character not the 0 & 1 integers.

Text

Description automatically generated

## Fix:

To fix that we subtract the ascii value of 0 to the value added to the vector of pins with that we get the value of 0 and 1 as integers, the ascii value of 0 is 48. It is also possible to subtract the character ‘0’ which is the same operation.

# Bug 6

The Lorenz Machine uses a specific way to cipher the message sent with a specific rotation of the wheels. In this case, in lorenzMachine.cpp line 21 and 23 are incorrect.

Text

Description automatically generated

## Fix:

Swap the value of line 21 and line 23.

# Bug 7

Using a global check approach, in teleprinter.cpp in line 92 there is an error. The encoding part works in the way that we pass a baudot code to the machine it encrypts using that code and gives an encrypted code to transform to a readable value. We are calling twice the getCharFromBaudot.

A screenshot of a computer

Description automatically generated with medium confidence

## Fix:

Apply the class function getBaudotFromChar() passing the one letter each time.

# Bug 8

Running the program, we have the correct value in the first character, but the rest are incorrect which means that there is or are errors in the rotation of the wheel. In line 8 in lorenzMachine.cpp we can see that we are assigning the incorrect value to m37 and m61.

Text

Description automatically generated

## Fix:

Swap the values assigned values from line 8 and 9.

# Bug 9

We can only enter one value and we want the possibility to enter more than one word separated by spaces. This light us to the error in line 21.

Text

Description automatically generated

## Fix:

We need to read the line with the std::getline()

# Bug 10

We still have and error with the rotations at some point the values start to become incorrect. This lead us to an error with the m61 and m37 in main.cpp in line 49 and 50.

Text

Description automatically generated

## Fix:

Swap the values where we assign the number of pins for each wheel in line 49 and 50.

# Bug 11

After running the program, we can still see some error with the rotations. In line 61, we are passing the values in a wrong order.

Text

Description automatically generated

## Fix:

Swap psi and chi in the encryptionDevice passing parameters.

# Bug 12

By running the program we can see that we have an error only in one character which leads us to line 27 in teleprinter.cpp which is actually the ‘R’ character that is the one giving us error. We have the wrong hexadecimal value.

A picture containing text

Description automatically generated

## Fix:

To fix it we change the value after assignment to 0x0a which is the value that matches the binary 01010