Lesson 49 CPP atoi & stoi

In this lesson, we want to make a program that prints the **binary of any number**, it will be as follows:

```
#include <iostream>
using namespace std;
int main() {
int x = 5;
for (int i = 31; i >= 0; i--)
//32 loop
if (x & (1 << i))
cout << "1";
else
cout << "0";
cout << endl;
}
```

output:

00000000000000000000000000000101

And if we want to print the negative number:

```
#include <iostream>
using namespace std;

int main() {
```

```
int x = ~5;
cout << x << endl;
for (int i = 31; i >= 0; i--)
if (x & (1 << i))
cout << "1";
else
cout << "0";
cout << endl;
}
```

output:

```
-6
11111111111111111111111111111010
```

If we want to convert a string to int :

```
#include <iostream>
#include <string>
using namespace std;

int main() {
string x = "2020";
int y = atoi (x.c_str());
//atoi works with C language
cout << y + 1 << endl;
```

}

output:

2021

But if we want to use the string in c++ without converting it, we will use **stoi :**

```
#include <string>
using namespace std;

int main() {
string x = "2020";
int y = stoi(x);
//stoi works with C++ language
cout << y + 1 << endl;
}
```

output:

2021

Note :

-There is a problem with stoi that if there is a character in the string it will have a problem with the code

- With atoi even if there is character it works well

But if we want to solve the problem of stoi, we use try and catch and its function if there is an error expected to occur,

but we want the program to continue working without stopping and we put the code inside them

```
#include <iostream>
#include <string>
using namespace std;

int main() {
string x = "%$^&#";
int y= -1;
try {
//we tell the computer to try this code
y = stoi(x);
}
catch (exception e) { // if the first part in try doesn't work this part will be executed

cout << "Error" << endl;
y = -1;
}
if(y!=-1)
cout << y + 1 << endl;
}
```

output:

Error