CMPSC 177 HW1 code

Brian Lee, Thurs 5pm

October 2017

1 Code

```
/**Brian Lee
 * 3620101
 * CMPSC 177
 * Brute Force Shift Cipher Breaking Code
 **/

#include <iostream>
#include <string>
#include <vector>

using namespace std;

int convertToNum(char c)
{
    if(c == 'a')
    {
        return 0;
    }
    else if(c == 'b')
    {
        return 1;
    }
    else if(c == 'c')
    {
        return 2;
    }
    else if(c == 'd')
    {
        return 3;
    }else if(c == 'e')
    {
        return 4;
```

```
}else if(c == 'f')
{
    return 5;
}else if(c == 'g')
{
    return 6;
}else if(c == 'h')
{
    return 7;
}else if(c == 'i')
{
    return 8;
}else if(c == 'j')
{
    return 9;
}else if(c == 'k')
{
    return 10;
}else if(c == 'l')
{
    return 11;
}else if(c == 'm')
{
    return 12;
}else if(c == 'n')
{
    return 13;
}else if(c == 'o')
{
    return 14;
}
else if (c =='p')
{
    return 15;
}
else if (c =='q')
{
    return 16;
}
else if (c =='r')
{
    return 17;
}
else if (c =='s')
{
    return 18;
```

```
    }
    else if (c =='t')
    {
        return 19;
    }
    else if (c =='u')
    {
        return 20;
    }
    else if (c =='v')
    {
        return 21;
    }
    else if (c =='w')
    {
        return 22;
    }
    else if (c =='x')
    {
        return 23;
    }
    else if (c =='y')
    {
        return 24;
    }
    else if (c =='z')
    {
        return 25;
    }
    else
    {
        cout << "Incorrect Value.\n";
        return -1;
    }
}

char convertToChar(int x)
{
    if(x == 0)
    {
        return 'a';
    }
    else if(x == 1)
    {
        return 'b';
    }
```

```
else if(x == 2)
{
    return 'c';
}
else if(x == 3)
{
    return 'd';
}
else if(x == 4)
{
    return 'e';
}
else if(x == 5)
{
    return 'f';
}
else if(x == 6)
{
    return 'g';
}
else if(x == 7)
{
    return 'h';
}
else if(x == 8)
{
    return 'i';
}
else if(x == 9)
{
    return 'j';
}
else if(x == 10)
{
    return 'k';
}
else if(x == 11)
{
    return 'l';
}
else if(x == 12)
{
    return 'm';
}
else if(x == 13)
{
```

```
    return 'n';
}
else if(x == 14)
{
    return 'o';
}
else if(x == 15)
{
    return 'p';
}
else if(x == 16)
{
    return 'q';
}
else if(x == 17)
{
    return 'r';
}
else if(x == 18)
{
    return 's';
}
else if(x == 19)
{
    return 't';
}
else if(x == 20)
{
    return 'u';
}
else if(x == 21)
{
    return 'v';
}
else if(x == 22)
{
    return 'w';
}
else if(x == 23)
{
    return 'x';
}
else if(x == 24)
{
    return 'y';
}
```

```
    else if(x == 25)
    {
        return 'z';
    }
    else
    {
        cout << "Invalid Number.\n";
        return ' ';
    }
}

int main()
{
    string input;
    string output = "";

    cout << "Input ciphertext: ";
    cin >> input;
    cout << endl;

    cout << "Deciphering code..." << endl;

    vector<int> numValues;
    //Convert ciphertext to nums
    for(string::iterator iter = input.begin(); iter != input.end(); ++iter)
    {
        int num = convertToNum(*iter);
        numValues.push_back(num);
    }

    /*
    for(int i = 0; i < numValues.size(); ++i)
    {
        cout << numValues.at(i) << " ";
    }
    cout << endl;
    */

    //Iterate through keys
    for(int k = 1; k < 27; ++k)
    {
        //Add key to numbers
        vector<int> newValues;
        for(int i = 0; i < numValues.size(); ++i)
        {
            int x = (numValues.at(i) + k) % 26;
```

```
            newValues.push_back(x);
        }

        //Convert new numbers to string
        for(int j = 0; j < newValues.size(); ++j)
        {
            char c = convertToChar(newValues.at(j));
            output += c;
        }
        cout << "Key: " << k << endl;
        cout << "Deciphered output: " << output << endl;

        output = "";
    }

    return 0;
}
```