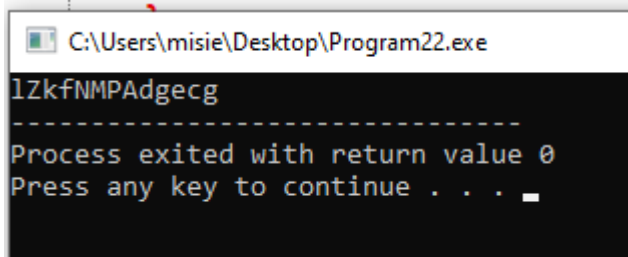


Program szyfrujący

Michał Jakub Krupa

Działający program:



```
C:\Users\misie\Desktop\Program22.exe
lZkfNMPAdgecg
-----
Process exited with return value 0
Press any key to continue . . .
```

Kod:

```
#include <iostream>
#include <string>
using namespace std;
void szyfruj(string text, string key) {
    string output;
    int dlugosc;
    int dlugosc2;
    bool ktory;
    // sprawdzanie dlugosci tekstow
    if (text.length() < key.length())
    {
        dlugosc = text.length();
        dlugosc2 = key.length();
        ktory = 0;
    }
    else {
        dlugosc = key.length();
        dlugosc2 = text.length();
        ktory = 1;
    }
    // Litery zamieniamy na intigery w przedziale od 65 do 122 tak jak w unicode (bez 92-
    96 poniewaz sa to znaki specjalne a nie litery alfabetu), dla spacji zamiast liczby 32
    zarezerwowalismy liczbe 91
    for (int i = 0; i < dlugosc ; i++)
    {
        if (((int)text[i] + (int)key[i] > 122)) {
```

```

        if (((((int)text[i] + (int)key[i]) % 122)) < 65) {
            if (((((int)text[i] + (int)key[i]) % 122) + 57) >= 91 && (((int)text[i]
+ (int)key[i]) % 122) + 57) <= 96)
            {
                if((((int)text[i] + (int)key[i]) % 122) + 57) == 91)
                    output += 32;
                else
                    output += (((int)text[i] + (int)key[i]) % 122) + 65 + 7;
            }
            else
            {
                output += (((int)text[i] + (int)key[i]) % 122) + 57;
            }
        }
        else {
            if (((((int)text[i] + (int)key[i]) % 122)) >= 91 && (((int)text[i] +
(int)key[i]) % 122) <= 96)
            {
                if((((int)text[i] + (int)key[i]) % 122)) == 91)
                    output += 32;
                else
                    output += (((int)text[i] + (int)key[i]) % 122) + 7;
            }
            else
                output += (((int)text[i] + (int)key[i]) % 122);
        }
    }
    else
    {
        if ((int)text[i] + (int)key[i] >= 92 && (int)text[i] + (int)key[i] <= 96)
        {
            if((int)text[i] + (int)key[i] == 92)
                output += 32;
            else
                output += (int)text[i] + (int)key[i] + 7;
        }
        else
        {
            if ((int)text[i] + (int)key[i] == 97)
                output += 32;
            else
                output += (int)text[i] + (int)key[i] + 1;
        }
    }
}
if (ktory == 0)
{
    for (int i = dlugosc; i < dlugosc2; i++)

```

```

        {
            output += key[i];
        }
    }
    else {
        for (int i = dlugosc; i < dlugosc2; i++)
        {
            output += text[i];
        }
    }
    cout<<output;
} //text na szyfr
int main() {
    string text = "text na testy";
    string klucz = "romeo i julia";
    szyfruj(text,klucz);
}

```

Pytania.

1. Wpisujemy każdą możliwą literę alfabetu - z małej i dużej litery. Zarówno w text jak i klucz.

```

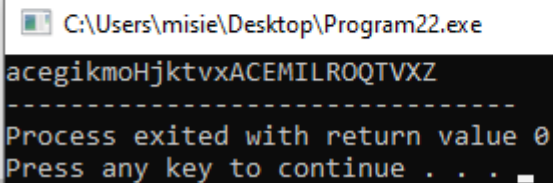
text = "ABCDEFGH IJKLMNOPRST UVWXYZ"
klucz = "abcdefghi jklmnoprstuvwxyz"

```

```

} //text na szyfr
int main() {
    string text = "ABCDEFGH IJKLMNOPRST UVWXYZ";
    string klucz = "abcdefghi jklmnoprstuvwxyz";
    szyfruj(text,klucz);
}

```



```

C:\Users\misie\Desktop\Program22.exe
acegikmoHjktvxACEMILROQTVXZ
-----
Process exited with return value 0
Press any key to continue . . .

```

2. Żeby nasz program miał działać jak szyfr Cezara, nasz klucz musi posiadać ciąg tych samych znaków. Np. duża litera A, przesunie nasze znaki o 65 pozycji w alfabecie.
3. W naszym programie do deszyfrowania tekstem jest nasz szyfr słowa "text na szyfr" - "lZkfNMPAdgecg". Jako klucz podstawiamy "ILNONbR QqORSI".

```
    cout << endl;
} //text na szyfr
int main() {
    string text = "lZkfNMPAdgecg";
    string klucz = "ILNONbR QqORS";
    szyfruj(text, klucz);
}
```

