

Nama : Raihan Hadi Nafi'T
Nim : 1900018287
Slot : Selasa 16.30

“PRETEST “

1. Kriptografi adalah ilmu yang mempelajari teknik enkripsi naskah asli (plaintext) yang tersusun acak, dengan memanfaatkan kunci enkripsi sehingga naskah tersebut berubah menjadi teks yang sulit terbaca (ciphertext) oleh user yang tidak memiliki kunci dekripsi.

2. Jenis Jenis Kriptografi :

- Symmetric key cryptography

Symmetric key cryptography atau disebut juga sebagai kriptografi kunci rahasia. Dalam symmetric key cryptography ini penerima dan pengirim informasi hanya menggunakan satu kunci untuk mengenkripsi dan mendeskripsikan pesan.

- Public key cryptography

Public key cryptography adalah konsep perlindungan data yang paling revolusioner dalam kurun waktu 300 hingga 400 tahun terakhir. Public key cryptography lebih dikenal dengan kriptografi kunci publik, metode kriptografi ini memanfaatkan dua kunci yang saling berkaitan, yaitu kunci publik dan privat.

- Hash function

Hash function adalah sistem kerja dari metode kriptografi, dengan meringkas informasi dan mengirimkan penjelasannya yang telah dirangkum

- 3.

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
| D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | A | B | C |

Plaintext : RAIHAN HADI NAFI MAHASISWA FAKULTAS TEKNOLOGI INDUSTRI TEKNIK
INFORMATIKA UNIVERSITAS AHMAD DAHLAN

ChiperText : UDLKDQ KDGL QDIL PDKDVLVZD IDNXOWDV WHNQRORJL LQGxVWUL
WHNLN LQIRUPDWLND XQLYHUVLWDV DKPDG GSJODQ

“Kegiatan “

```
#include <iostream>
#include <string.h>
#define maks 500
using namespace std;

int main(){
    char chiper[maks];
    char plain[maks];
    int key;
    int jumlah=0;

    cout<<"Masukkan kata : ";
    cin.getline(chiper, sizeof(chiper));

    cout<<"Masukkan key : ";
    cin>>key;

    // berikut adalah langkah untuk merubah plaintext menjadi chipper text
    for(int i=0; i<strlen(chiper); i++){
        cout<<chiper[i] <<"( "<<int(chiper[i])<<" )"; //ini merupakan fungsi untuk manggil code
        asciper
        jumlah=jumlah+int(chiper[i]);
        chiper[i] =(chiper[i]+key)%128;

    }

    cout<<endl;
    cout<<"total = "<<jumlah;

    cout<<"\nhasil enkripsi : "; //ini merupakan hasil untuk manggil kode setelah di enkripsi
    for(int i = 0; i<strlen(chiper); i++){
        if(chiper[i]>=90){
            //berikut merupakan pemberian aturan pad koding
            //Agar ketika Melebihi 90 Di kurang dengan 26 supaya tidak lebih dari 90
            //jika melebihi maka berbeda dengan hasil manual, karena ini berdasarkan tabel
            ascii
            chiper[i]=chiper[i]-26;
            cout<<chiper[i];
        }
        else{
            cout<<chiper[i];
        }
    }
}
```

```
}
```

```
//Ini merupakan cara untuk merubah chipper text menjadi plaintext kembali
```

```
for(int i=0; i<strlen(chiper); i++){ //ubah kembali enkripsi ke dekripsi
```

```
    plain[i]=(chiper[i]-key)%128;
```

```
    chiper[i]=plain[i];
```

```
}
```

```
cout<<"\nhasil dekripsi : "; //Ini fungsi untuk memanggil Kode setelah di dekripsi
```

```
for(int i=0; i<strlen(chiper); i++){
```

```
    if(chiper[i]<=64){
```

```
        // memberikan aturann :
```

//Jika kurang dari 64 maka akan di tambah dengan 26 supaya angka tidak kurang dari 64 dan dapat kembali ke plaintext awal,

//jika kurang dari 64 maka dia tidak akan sama dengan manual karena ini berdasarkan tabel ascii

```
        chiper[i]=chiper[i]+26;
```

```
        cout<<chiper[i];
```

```
    }
```

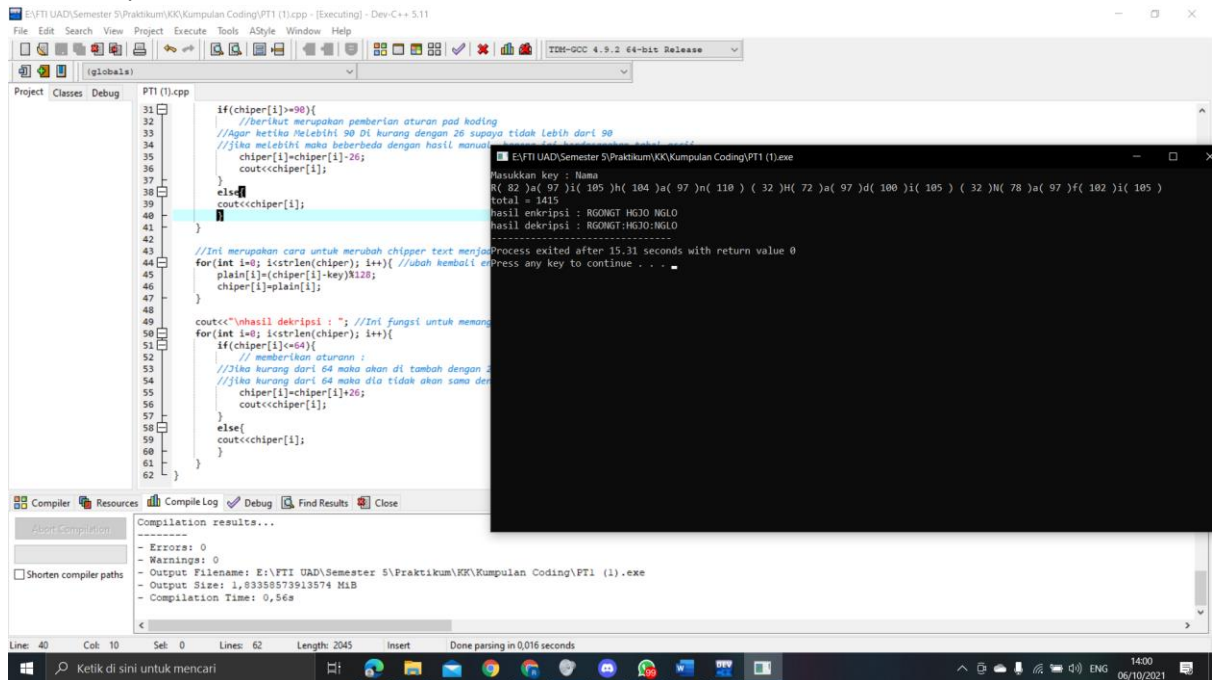
```
    else{
```

```
        cout<<chiper[i];
```

```
    }
```

```
}
```

Hasil Compile :



“POSTEST”

1. Plaintext : TEKNIK INFORMATIKA FTI UNIVERSITAS AHMAD DAHLAN YOGYAKARTA

Key : GADALAWAN

| A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 |
| | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | |

| | | | | | | |
|------------|-------|------|-------|-------|-------|-------|
| PLAIN | T(19) | E(4) | K(10) | N(13) | I(8) | K(10) |
| KEY | G(6) | A(0) | D(3) | A(0) | L(11) | A(0) |
| P+k(mod26) | 25 | 4 | 13 | 13 | 19 | 10 |
| CHIPER | Z | E | N | N | T | K |

| | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|---|----|---|----|---|----|----|---|
| PLAIN | I | N | F | O | R | M | A | T | I | K | A | F | T | I |
| | 8 | 13 | 5 | 14 | 17 | 12 | 0 | 19 | 8 | 19 | 0 | 5 | 19 | 8 |
| KEY | W | A | N | G | A | D | A | L | A | W | A | N | G | A |
| | 22 | 0 | 13 | 6 | 0 | 3 | 0 | 11 | 0 | 22 | 0 | 13 | 6 | 0 |
| P+k(mod26) | 4 | 13 | 18 | 20 | 17 | 15 | 0 | 4 | 8 | 6 | 0 | 18 | 25 | 8 |
| CHIPER | E | N | S | U | R | P | A | E | I | G | A | S | Z | I |

2.

Coding Enscripsi Descripsi dengan metode Vigenere :

```
#include <iostream>
#include <string>
#include<cstdlib>
#define maks 500

using namespace std;

class Vigenere {

public:
    string key;
    Vigenere(string key)
    {
        for(int i = 0; i < key.size();++i)
        {
```

```

        if (key[i] >= 'A' && key[i] <= 'Z')
            this->key += key[i];
        else if (key[i] >= 'a' && key[i] <= 'z')
            this->key += key[i] + 'A' - 'a';
    }
}

string encrypt(string text)
{
    string out;

    for (int i = 0, j = 0; i < text.length(); ++i)
    {
        char c = text[i];

        if (c >= 'a' && c <= 'z')
            c += 'A' - 'a';
        else if (c < 'A' || c > 'Z')
            continue;

        out += (c + key[j] - 2 * 'A') % 26 + 'A';
        j = (j + 1) % key.length();
    }

    return out;
}

string decrypt(string text)
{
    string out;

    for (int i = 0, j = 0; i < text.length(); ++i)
    {
        char c = text[i];

        if (c >= 'a' && c <= 'z')
            c += 'A' - 'a';
        else if (c < 'A' || c > 'Z')
            continue;

        out += (c - key[j] + 26) % 26 + 'A';
        j = (j + 1) % key.length();
    }

    return out;
}

};

int main(){
    string a,b;
    cout<<"Masukkan Plaintext : ";

```

```

cin>>a;
cout<<"Masukkan Key : ";
cin>>b;
string original=a;
Vigenere chiper=b;

string encrypted = chiper.encrypt(original);
string decrypted = chiper.decrypt(encrypted);
cout<<endl;
cout<<"encrypted : "<<encrypted<<endl;
cout<<"decrypted : "<<decrypted<<endl;
}

```

Hasil Compile coding diatas :

