信息安全作业（一）

计算机 1202　张艺瀚
学号:20123852

March 27, 2015

程序使用说明：在 Linux 下当前目录执行可执行文件，按照提示操作即可，具体见运行结果截图。

1　编程实现双轨算法

1.1　代码清单

```
int main(int argc, char** argv){
  cout << "Input plaintext: ";
  string plaintext, ciphertext;
  cin >> plaintext;
  int t = floor((plaintext.size() - 1) / 2);
  for(unsigned int i = 0; i < plaintext.size(); ++i){
    if(i <= t){
      ciphertext.push_back(plaintext[2 * i]);
    }else{
      ciphertext.push_back(plaintext[2 * (i - t) - 1]);
    }
  }
  cout << "Ciphertext is: " << ciphertext << endl;
  return 0;
}
```

Listing 1: 双轨算法的 C++ 实现

1.2　说明

从键盘读入用户的明文，输出密文。

若同一字符在明文中的下标为 i，在密文中的下标为 i'，则有

$$i = \begin{cases} 2i', & i' \leq \lfloor \frac{l-1}{2} \rfloor \\ 2\left(i' - \lfloor \frac{l-1}{2} \rfloor\right) - 1, & else \end{cases}$$

其中 $l = len(plaintext)$

1.3 运行结果

```
zephyr@ubuntu:~/code/cpp/information_security/1/1.1$ ./1-1
Input plaintext: discreteandsystem
Ciphertext is: dsrtadytmiceensse
```

Figure 1: 双轨算法的运行结果

2 编程实现钥控算法

2.1 代码清单

```
template<typename T>
void display(const vector<T>& v){
  for_each(v.begin(), v.end(),
    [](T t){
      cout << t << " ";
    });
  cout << endl;
}

int main(int argc, char** argv){
  string plaintext, ciphertext, key;
  cout << "Input plaintext: ";
  cin >> plaintext;
  cout << "Input key: ";
  cin >> key;
  int dist = 26;
  vector<int> v(key.size());
  for(size_t i = 0; i < v.size(); ++i){
    v[i] = key[i] - 'a';
```

```
  }
  int r = plaintext.size() / key.size();
  if(r * key.size() != plaintext.size()){
    ++r;
  }
  int n = r * key.size();
  for(size_t i = 0; i < key.size(); ++i){
    int m = 26, idx;
    for(size_t j = 0; j < v.size(); ++j){
      if(v[j] >= 0 && v[j] < m){
        idx = j;
        m = v[j];
      }
    }
    v[idx] = -1;
    for(auto p = plaintext.begin() + idx;;){
      int d = p - plaintext.begin();
      if(d < plaintext.size()){
        ciphertext.push_back(*p);
      }else if(d < n){
        ciphertext.push_back('z');
      }
      if(d < n){
        p += key.size();
      }else{
        break;
      }
    }
  }
  cout << "Ciphertext is: " << ciphertext << endl;
  return 0;
}
```

Listing 2: 钥控算法的 C++ 实现

2.2 说明

从键盘读入用户的明文和密钥，输出密文。

若 $l = len(key)$，将明文写成一个 l 列矩阵，按照 key 中个字符的字典序将每一列写出即为密文。

2.3 运行结果

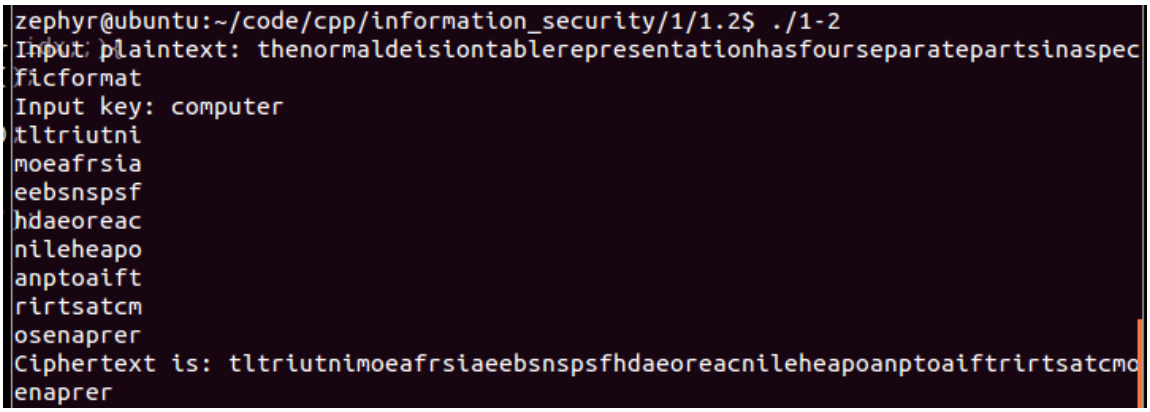

Figure 2: 钥控算法的运行结果