Final Year B. Tech, Sem VII 2022-23
PRN – 2020BTECS00211
Name – Aashita Narendra Gupta
Cryptography And Network Security Lab
Batch: B4

Practical No – 3

Title: To implement Playfair Cipher Cipher.

Theory:
Playfair cipher is an encryption algorithm to encrypt or encode a message. It is the same as a traditional cipher. The only difference is that it encrypts a digraph (a pair of two letters) instead of a single letter.
It initially creates a key-table of 5*5 matrix. The matrix contains alphabets that act as the key for encryption of the plaintext. Note that any alphabet should not be repeated. Another point to note that there are 26 alphabets and we have only 25 blocks to put a letter inside it. Therefore, one letter is excess so, a letter will be omitted (usually J) from the matrix. Nevertheless, the plaintext contains J, then J is replaced by I. It means treat I and J as the same letter, accordingly.
Since Playfair cipher encrypts the message digraph by digraph. Therefore, the Playfair cipher is an example of a digraph substitution cipher.

Example:
Suppose, the plaintext is COMMUNICATION and the key that we will use to encipher the plaintext is COMPUTER.

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

Therefore, the plaintext COMMUNICATE gets encipher (encrypted) into OMRMPCSGPTER.

Code Snapshots:

```
#include<iostream>
#include<string>
#include<vector>
#include<map>
using namespace std;
int main(){
    int i,j,k,n;
    cout<<"Enter the message: ";
    string s,origin;
    getline(cin,origin);
    cout<<"Enter the key: ";
    string key;
    cin>>key;
    int choice;

    cout<<"Enter your choice(1 OR 2)."<<endl;
    cout<<"1. Encryption"<<endl;
    cout<<"2. Decryption"<<endl;
    cout<<"Your choice: ";
    cin>>choice;
    if(choice==1)
    {
    for(i=0;i<origin.size();i++){
        if(origin[i]!=' ')
            s+= origin[i];
    }
    vector<vector<char> > a(5,vector<char>(5,' '));
    n=5;
    map<char,int> mp;
    k=0;
    int pi,pj;
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            while(mp[key[k]]>0&&k<key.size()){
                k++;
            }
            if(k<key.size()){
                a[i][j]=key[k];
                mp[key[k]]++;
                pi=i;
                pj=j;
            }
            if(k==key.size())
            break;
        }
        if(k==key.size())
            break;
```

```
    }
    k=0;
    for(;i<n;i++){
        for(;j<n;j++){
            while(mp[char(k+'a')]>0&&k<26){
                k++;
            }
            if(char(k+'a')=='j'){
                j--;
                k++;
                continue;
            }
            if(k<26){
                a[i][j]=char(k+'a');
                mp[char(k+'a')]++;
            }
        }
        j=0;
    }
    string ans;
    if(s.size()%2==1)
        s+="x";
    for(i=0;i<s.size()-1;i++){
        if(s[i]==s[i+1])
            s[i+1]='x';
    }
    map<char,pair<int,int> > mp2;
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            mp2[a[i][j]] = make_pair(i,j);
        }
    }

    for(i=0;i<s.size()-1;i+=2){
        int y1 = mp2[s[i]].first;
        int x1 = mp2[s[i]].second;
        int y2 = mp2[s[i+1]].first;
        int x2 = mp2[s[i+1]].second;
        if(y1==y2){
            ans+=a[y1][(x1+1)%5];
            ans+=a[y1][(x2+1)%5];
        }
        else if(x1==x2){
            ans+=a[(y1+1)%5][x1];
            ans+=a[(y2+1)%5][x2];
        }
        else {
            ans+=a[y1][x2];
```

```
            ans+=a[y2][x1];
        }
    }
    cout<<ans<<'\n';
    }



    if(choice==2)
    {
        vector<vector<char> > a(5,vector<char>(5,' '));
    n=5;
    map<char,int> mp;
    k=0;
    int pi,pj;
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            while(mp[key[k]]>0&&k<key.size()){
                k++;
            }
            if(k<key.size()){
                a[i][j]=key[k];
                mp[key[k]]++;
                pi=i;
                pj=j;
            }
            if(k==key.size())
            break;
        }
        if(k==key.size())
            break;
    }
    k=0;
    for(;i<n;i++){
        for(;j<n;j++){
            while(mp[char(k+'a')]>0&&k<26){
                k++;
            }
            if(char(k+'a')=='j'){
                j--;
                k++;
                continue;
            }
            if(k<26){
                a[i][j]=char(k+'a');
                mp[char(k+'a')]++;
            }
        }
        j=0;
```

```
    }
    string ans;
    map<char,pair<int,int> > mp2;
    for(i=0;i<n;i++){
        for(j=0;j<n;j++){
            mp2[a[i][j]] = make_pair(i,j);
        }
    }
    for(i=0;i<origin.size()-1;i+=2){
        int y1 = mp2[origin[i]].first;
        int x1 = mp2[origin[i]].second;
        int y2 = mp2[origin[i+1]].first;
        int x2 = mp2[origin[i+1]].second;
        if(y1==y2){
            ans+=a[y1][(x1-1)%5];
            ans+=a[y1][(x2-1)%5];
        }
        else if(x1==x2){
            ans+=a[(y1-1)%5][x1];
            ans+=a[(y2-1)%5][x2];
        }
        else {
            ans+=a[y1][x2];
            ans+=a[y2][x1];
        }
    }
    if(ans[ans.size()-1]=='x')
        ans[ans.size()-1]='\0';
    for(i=1;i<ans.size();i++){
        if(ans[i]=='x')
            ans[i]=ans[i-1];
    }

    cout<<ans<<'\n';
    }
    return 0;
}
```

Output Snapshots:

Encryption:

```
PS C:\Users\Ashitra\OneDrive\Desktop\7th sem\Practicals\CNS\Programs> cd "c:\Users\Ashitra\OneDrive
\Desktop\7th sem\Practicals\CNS\Programs\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunner
File } ; if ($?) { .\tempCodeRunnerFile }
Enter the message: Communication
Enter the key: Computer
Enter your choice(1 OR 2).
1. Encryption
2. Decryption
Your choice: 1
ompwpqsibekCxp
```

Decryption:

```
PS C:\Users\Ashitra\OneDrive\Desktop\7th sem\Practicals\CNS\Programs> cd "c:\Users\Ashitra\OneDrive
\Desktop\7th sem\Practicals\CNS\Programs\" ; if ($?) { g++ tempCodeRunnerFile.cpp -o tempCodeRunner
File } ; if ($?) { .\tempCodeRunnerFile }
Enter the message: ompwpqsibekCxp
Enter the key: Computer
Enter your choice(1 OR 2).
1. Encryption
2. Decryption
Your choice: 2
Communication
PS C:\Users\Ashitra\OneDrive\Desktop\7th sem\Practicals\CNS\Programs>
```

Conclusion:

1. Playfair cipher is a symmetric encryption technique which is rich enough to encrypt all alphabets, numerals and most commonly used special symbols.
2. It uses trigraph rather than using digraph to eliminate the fact that a diagram and its reverse will encrypt in a similar fashion.
3. It is comparatively stronger algorithm.