Playfair Chiper

Tatiana Moreira Beita

Nanjing University of Information Science and Techonology

This project is an implementation of the playfair chiper algorithm, it is developed in the programming language scala and oriented to functional programming paradigm.

The algorithm follows the next steps:

Working with the keyword

1. The user input the keyword

```
// readLine lets you prompt the user and read their input
 as a String
   val key = readLine("What's the key? ")
```

2. With the “makeCompatible” function the key becomes compatible for future processes, this function is used to make compatible other strings too.

```
def makeCompatible(s: String) = s toUpperCase() replace('
J', 'I') filter(_.isLetter)
val keyUpper = makeCompatible(key)
```

3. In this project the key matrix is treated as a list

```
val intoList = keyUpper.toList
```

4. The repeated characters are extracted from the keyword.

```
val noRepeatedKeyWord = intoList.distinct
```

5. The alphabet is defined and treatened in order to make it compatible and we transformed the string into a list.

```
val abc = "abcdefghijklmnopqrstuvwxyz"
val upperAbc = makeCompatible(abc)
val listAbc = upperAbc.toList
```

6. We cancat the keyword and the alphabet to make the key matrix

```
val matrix = noRepeatedKeyWord ::: listAbc
```

7. At this point your list contains repeated elements, so let’s clean it

```
val noRepeatedMatrix = matrix.distinct
```

Working with the PT