Project 1-1: 6轮DES的编程实现

14307130356 卢颖

1. 目录结构

```
+ DES
├── README.md                    # 说明文档
├── 说明文档.pdf                  # 说明文档(和README.md内容相同)
├── index.html                   # DES主页
├── html                         # 网页
│   └── des.html                 # DES加解密界面
├── css                          # 网页样式
│   ├── bootstrap.min.css
│   └── common.css
├── libs                         # 一些js库
│   ├── bootstrap.min.js
│   ├── jquery.min.js
│   └── d3.min.js
├── testcase                     # 测试样例
│   ├── key.txt                  # 加解密所用密钥
│   ├── plain.txt                # 样例明文，cipher.txt 用key.txt中密钥解密结果
│   └── cipher.txt               # 样例密文，plain.txt 用 key.txt中密钥加密的一次
结果
└── src                          # 源文件
    ├── main.js                  # 检查一些非法情况，提示加密成功
    ├── cbc_des.js               # CBC模式加密
    └── des_encrypt.js           # DES算法
```

2. 使用说明

点击index.html进入主页，点击左侧导航栏上的 `加解密` 按钮可以进入DES加解密界面

2.1 输入

- 选择加密/解密模式
- 64位明文/密文文件，可以通过目录选择本地文件
- 64位密钥，采用16位十六进制的方法进行输入，字母大小写均可

2.2 输出

- 加密/解密后所得的密文/明文
- 点击 `下载结果` 按钮可以将结果输出为二进制的txt格式文件，并保存到本地。

2.3 提示

- 密钥非法提示（过长、过短、出现非法字符）
- 上传文件无效提示（文件内出现非法字符）
- 加密/解密成功提示
- 解密失败提示（密文长度不合法）

3. 实现说明

1. PJ采用了6轮DES的加解密算法
2. 采用**CBC**模式进行加解密

- 加密：最后一组不足位时**随机**选取0/1**填充**，最后8位作为填充指示符，表示填充的总**bit**数（含8位自身）
- 解密：DES解密后根据最后8位移除填充字符得到明文