

# RELAZIONE CIFRARIO DI PLAYFAIR

## MICHELE MARTINI 104941

Il programma implementa il cifrario di Playfair prendendo i dati da file e creandone altri per salvare i risultati. I dati necessari per il cifrario, quindi alfabeto, carattere speciale e mancante e chiave sono presenti in un file `keyfile.txt`; mentre il messaggio è presente nel file `message.txt`. Il messaggio cifrato viene salvato nel file `encrypted_message.pf` all'interno della cartella `outputdir`. Il messaggio decifrato viene salvato nel file `decrypted_message.dec` all'interno della stessa cartella.

Per l'implementazione del cifrario ho utilizzato una struttura dati chiamata **data\_block** contenente:

1. **Alfabeto**(array di char lungo 25 caratteri)
2. **Carattere mancante**(char)
3. **Carattere speciale**(char)
4. **Chiave**(char\*)

Ho inoltre creato due file header e i loro rispettivi file sorgente:

- **FileHandler.h**: contiene funzioni per la gestione dei file, dalla lettura alla creazione, le funzioni principali sono:
  - **create\_data**: legge i dati dal file e popola la struct `data_block`
  - **encode\_file**: crea il file `.pf` dove viene salvata la stringa codificata
  - **decode\_file**: crea il file `.dec` dove viene salvata la stringa decodificata
  - **read\_message**: legge dal file il messaggio da criptare
  - **read\_encoded\_message**: legge dal file il messaggio criptato
- **FileHandler.c**:
  - **open\_file**: funzione che apre un file
  - **get\_data**: restituisce il `data_block` creato dal file
  - **adjust\_message**: aumenta memoria allocata del messaggio nel caso in cui non fosse sufficiente
- **Playfair.h**: contiene solo le funzioni **encode** e **decode** utilizzate nel main
- **Playfair.c**: contiene le funzioni necessarie per criptare o decriptare il messaggio:
  - **remove\_spaces**: rimuove gli spazi da una stringa
  - **generate\_key\_table**: si occupa di popolare la matrice di criptazione secondo le regole di playfair
  - **standardize\_string**: divide la stringa in coppie di lettere

- **encode\_string**: codifica la stringa usando la matrice e le regole di playfair
- **decode\_string**: decodifica la stringa usando la matrice e le regole di playfair
- **encode**: usa le altre funzioni per codificare un messaggio e salvarlo nel relativo file
- **decode**: usa le altre funzioni per decodificare un messaggio e salvarlo nel relativo file