## Úkol 3

Vytvořte program, který bude provádět jednoduché symetrické šifrování pomocí algoritmu postaveném na **Feistelově síti**. Program bude přijímat dva argumenty. První argument je vstupní soubor (např. `input.txt`) a druhý je soubor s klíči (`keys.txt`). Program načte vstupní soubor do paměti, rozdělí na bloky po 8 bitech (jeden znak ASCII) a provede potřebný počet iterací Feistelovy sítě. Předpis pro funkci *f* je uveden níže.

Program provede i dešifrování. Z důvodu lepší čitelnosti a kontroly je **nutné** zašifrované výstupy převést na hexadecimální podobu:
např: *57 69 6b 69 70 65 64 ff*
Dodržte prosím tento formát (tj. včetně mezer)

Soubor `keys.txt` obsahuje klíč. Každý řádek obsahuje část klíče ($K_0$ – $K_n$). Šifrování bude probíhat po blocích o velikosti 1B (1 ASCII znak == 8 bitů) a váš program bude provádět tolik iterací, kolik klíčů bude v souboru `keys.txt` (tzn. $K_0$ – $K_7$ = celkem 8 iterací v rámci jednoho bloku). Zajistěte, aby váš program fungoval i když bude soubor `keys.txt` obsahovat jiný počet podklíčů (např. klíče $K_0$ – $K_3$).

## Předpis funkce *f*: $\mathbf{f(R_{i-1}, K_i) = (R_{i-1} \text{ xor } K_i) \text{ and } \neg K_i}$

Výstupem programu bude soubor `output.txt` se třemi řádky v následujícím formátu:

*Hexadecimální reprezentace zašifrovaného vstupu*
*Hexadecimální reprezentace zpětně dešifrovaného vstupu*
*Textová čitelná podoba zpětně dešifrovaného vstupu*

Pokud šifrování a dešifrování proběhlo v pořádku, měl by se dešifrovaný text shodovat se souborem `input.txt`.

Základní verze programu musí umět korektně zašifrovat textový soubor `input.txt` (4 body). Další dva body můžete získat za možnost zašifrování libovolného (i binárního) souboru (např. `dwarf_small.bmp`, který je přiložen v archivu).

V případě netextového (binárního) vstupu bude výsledkem zpětného chodu rozšifrovaný soubor, který se musí shodovat s původním. Dešifrovanému souboru přidejte předponu `feistel_`. V případě textového vstupního souboru nemusíte vytvářet žádný další soubor.

### Hodnocení:

Max. počet bodů: **6**

Příklad spuštění:

```
java -jar feistel_BIT.jar input.txt keys.txt
//vytvořen soubor output.txt s formátem dle zadání

java -jar feistel_BIT.jar dwarf_small.bmp keys.txt
//vytvořen soubor output.txt s formátem dle zadání + soubor
feistel_dwarf_small.bmp
```

Odevzdání:

Vytvořte a odevzdejte archiv `BIT_ukol3_<jméno_prijmeni>.zip`. Součástí archivu budou zdrojové kódy programu a spustitelný soubor a také soubor `README.txt`. V něm stručně popište vaše řešení a případná omezení.

**Řádný termín odevzdání: 8.4.2021 23:59:59**

**Meziní termín odevzdání: 15.4.2021 23:59:59**