**Sprawozdanie**

*Bezpieczeństwo Sieci Komputerowych*

Pracownia Specjalistyczna 6-7



Temat:   
**KRYPTOSYSTEMY SYMETRYCZNE   
– ALGORYTM DES.**

Wykonanie:

**Busłowski Tomasz**

**Suchwałko Tomasz**

Prowadzący zajęcia: **dr inż. Maciej Brzozowski**

**Zadanie do wykonania**

Wykonaj program realizujący szyfrowanie oraz deszyfrowanie z wykorzystaniem algorytmu DES. Więcej informacji w dokumencie źródłowym fips46-3.

**Środowisko, framework i język implementacji zadań:**

* Microsoft Visual Studio Enterprise 2015 (Version 14.0.25431.01 Update 3).
* Microsoft .NET Framework (Version 4.6.01586).
* C# 6.0.

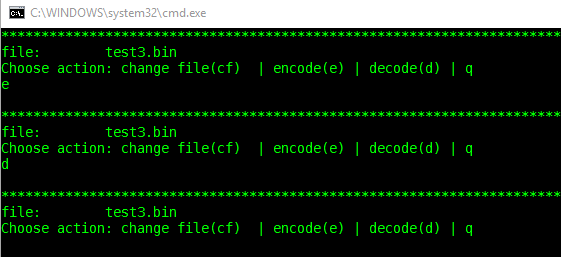
**Wykonanie zadań:**

Wszystkie zadania zostały wykonane.

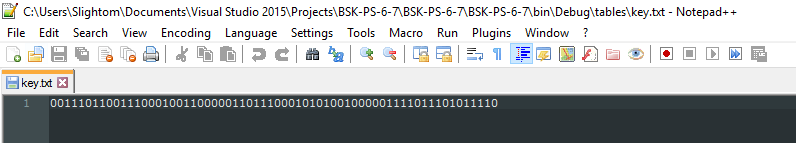
* Tomasz Busłowski – szyfrowanie & deszyfrowanie
* Tomasz Suchwałko – szyfrowanie & deszyfrowanie

**Screeny wykonanych zadań:**

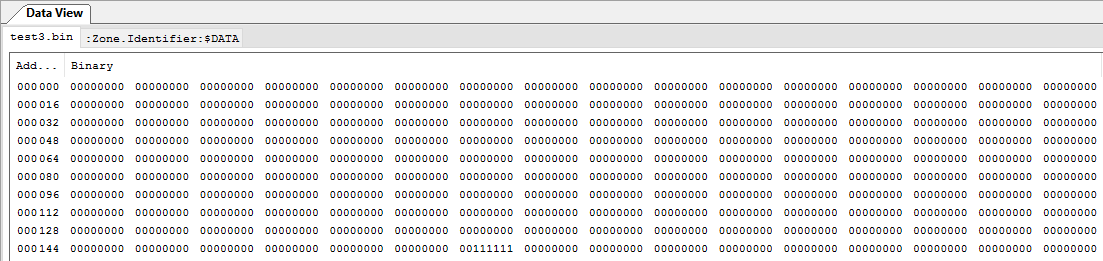
Przykład szyfrowanie i deszyrowania wiadomości naszym programem przy wykorzystaniu algorytmu DES:



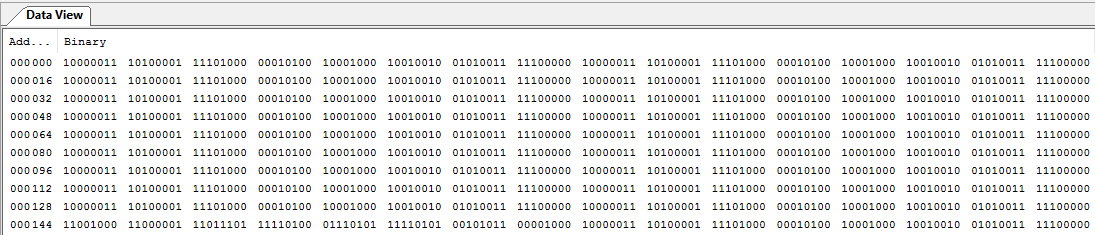
Rysunek 1(konsola, szyfrowanie i deszyfrowanie pliku test3.bin)



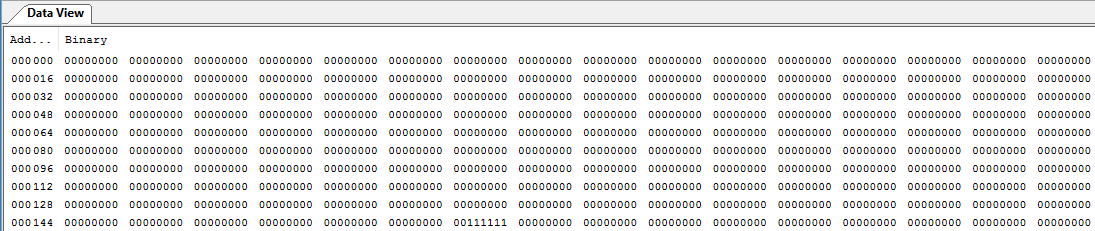
Rysunek 2(plik key.txt)



Rysunek 3(test3.bin, przed zaszyfrowaniem)



Rysunek 4(test3OUT.bin, zaszyfrowany plik test3.bin)



Rysunek 5 (test3OUT.bin, po rozszyfrowaniu)

Algorytm DES do poprawnego działania potrzebuje plików o wielokrotności 64 bitów(8bajtów). Pliki mają różne rozmiary, nie mamy gwarancji odpowiedniej wielkości pliku. Problem ten rozwiązaliśmy stosując padding pliku, który chcemy zaszyfrować. Działa on w taki sposób: jeżeli plik nie jest wielokrotnością 8 bajtów, to dopisujemy dodatkowe bajty o randomoweych wartościach tak aby otrzymać wielokrotność 8 bajtów a w ostatnim bajcie zapisujemy informacje ile bajtów dodaliśmy(ile mamy zignorować przy deszyfracji). Jeśli natomiast plik jest wielokrotnością 8 bajtów, to dodajemy   
7 randomowych bajtów i w 8 zapisujemy wartość 8.