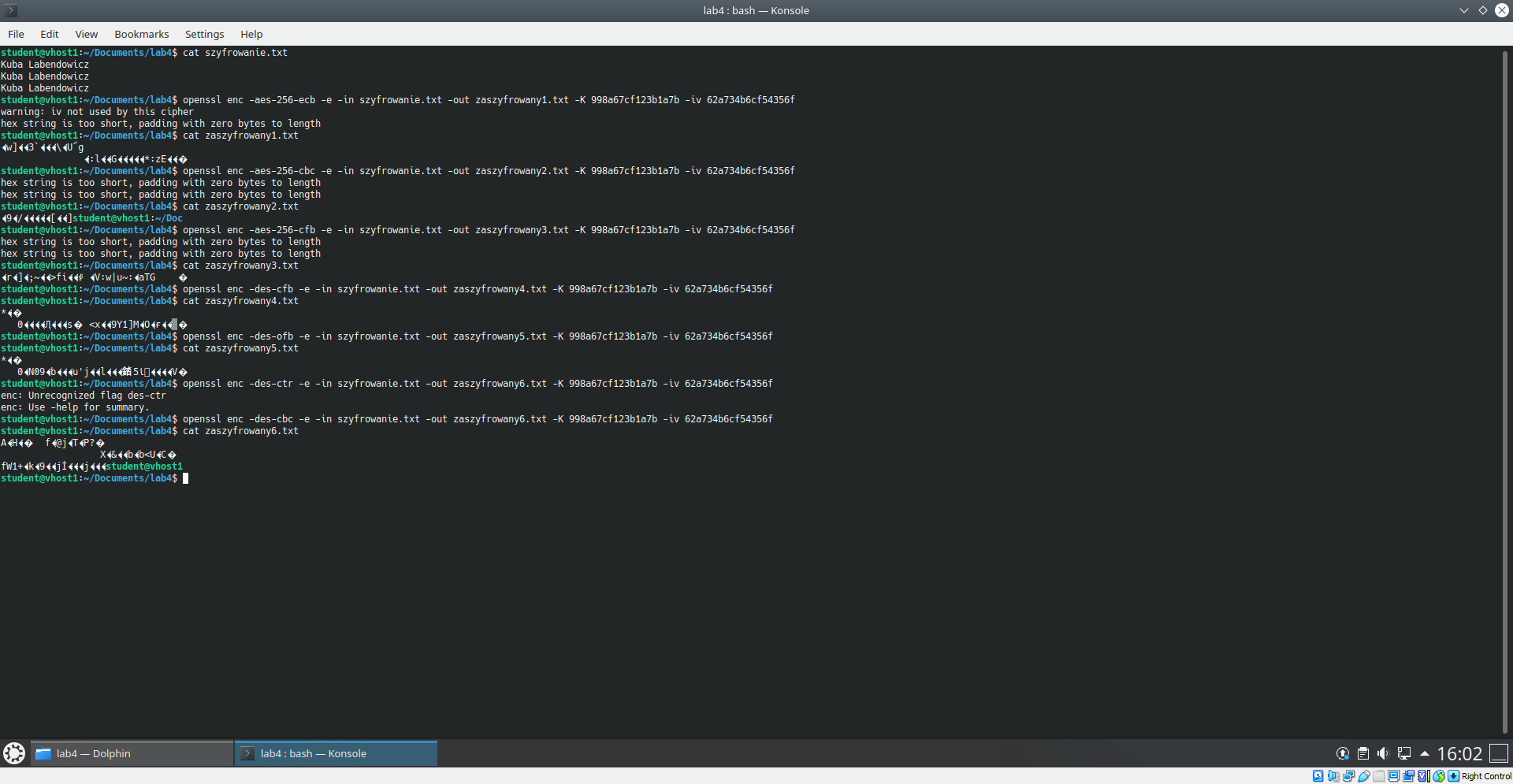
Sprawozdanie

LABORATORIUM 4. KRYPTOGRAFIA SYMETRYCZNA, TRYBY PRACY ALGORYTMÓW KRYPTOGRAFICZNYCH.

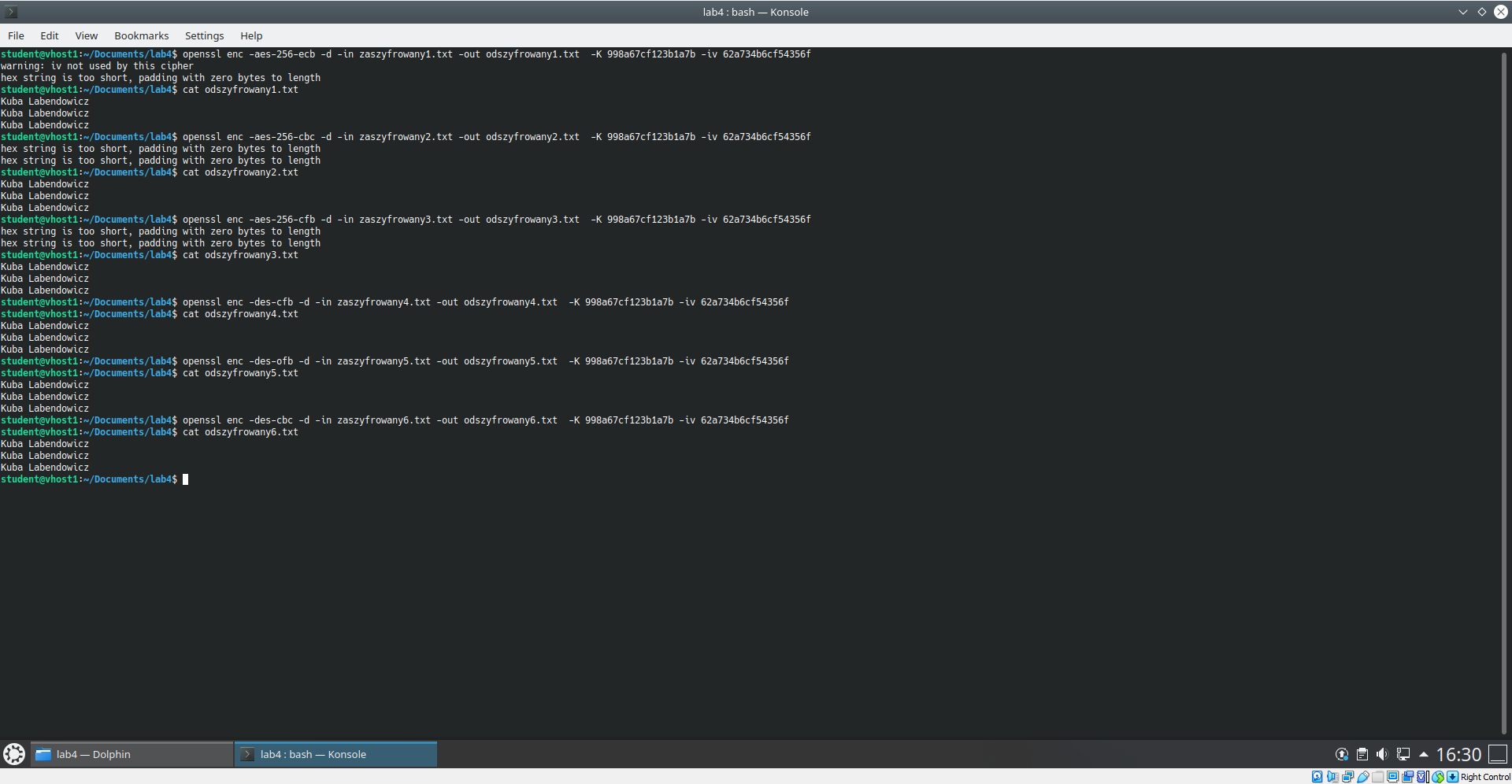
# Zadanie 4.1

## P.4.1

Przebieg szyfrowania



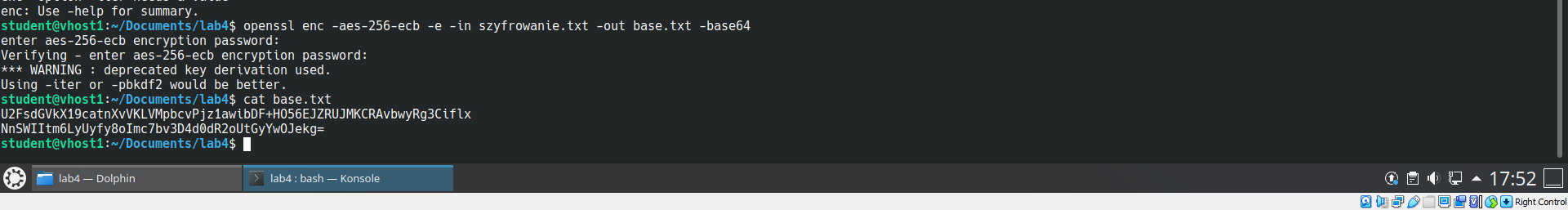
Przebieg odszyfrowywania



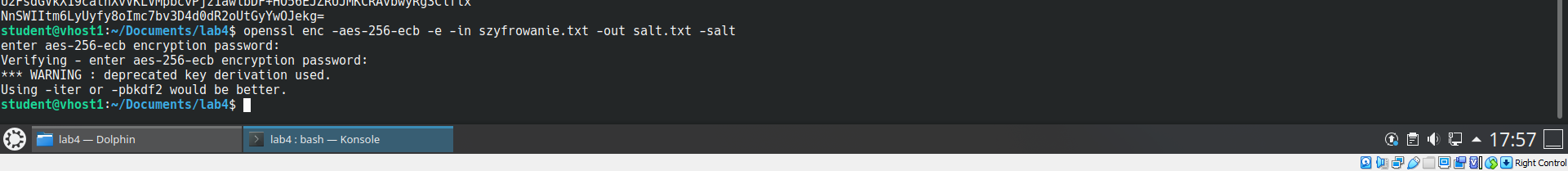
1. „hex string is too short, padding with zero bytes to length” oznacza, że podany klucz jest zbyt krótki i jest on wypełniany dodatkowymi zerami
2. Występuje komunikat „warning: iv not used by this cipher”, zatem nie wszystkie tryby wymagają wektora inicjalizującego

## P.4.2

Przebieg szyfrowania



Przebieg szyfrowania



1. Opcja –base64 koduje dane po zaszyfrowaniu.
2. Opcja –salt dodaje do klucza losowe znaki. Celem soli jest ochrona przed atakami słownikowymi.

## D.4.1

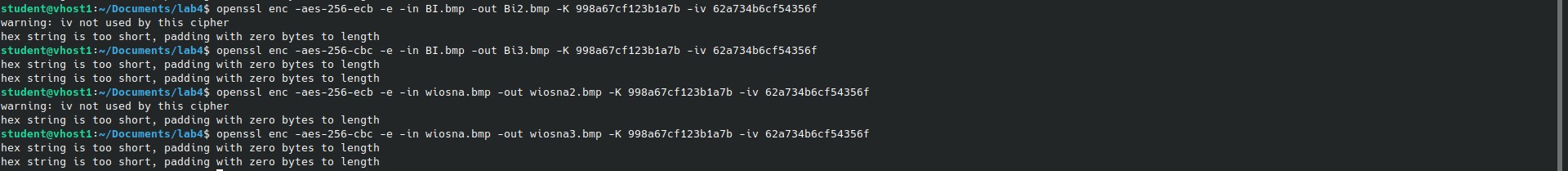
$ openssl enc -aes-256-cbc -md sha512 -pbkdf2 -iter 1000 -salt -in <plik\_wej> -out <plik\_wj>

|  |  |
| --- | --- |
| -aes-256-cbc | Typ szyfru aes-256 z trybem cbc |
| -md sha512 | Umożliwia utworzenie klucza z frazy hasła. Domyślnym algorytmem jest sha-256. |
| -pbkdf2 | Umożliwia użycie algorytmu pbkdf2 |
| -iter 1000 | Umożliwia użycie określonej liczby iteracji na haśle w celu uzyskania klucza szyfrowania. Wysokie wartości zwiększają czas wymagany do brute-force'owania pliku wynikowego. Opcja umożliwia użycie algorytmu PBKDF2 do uzyskania klucza. |
| -salt | Dodaje do klucza losowe znaki. Celem soli jest ochrona przed atakami słownikowymi. |
| -in <plik\_wej> | Plik wejściowy |
| -out <plik\_wj> | Plik wyjściowy |

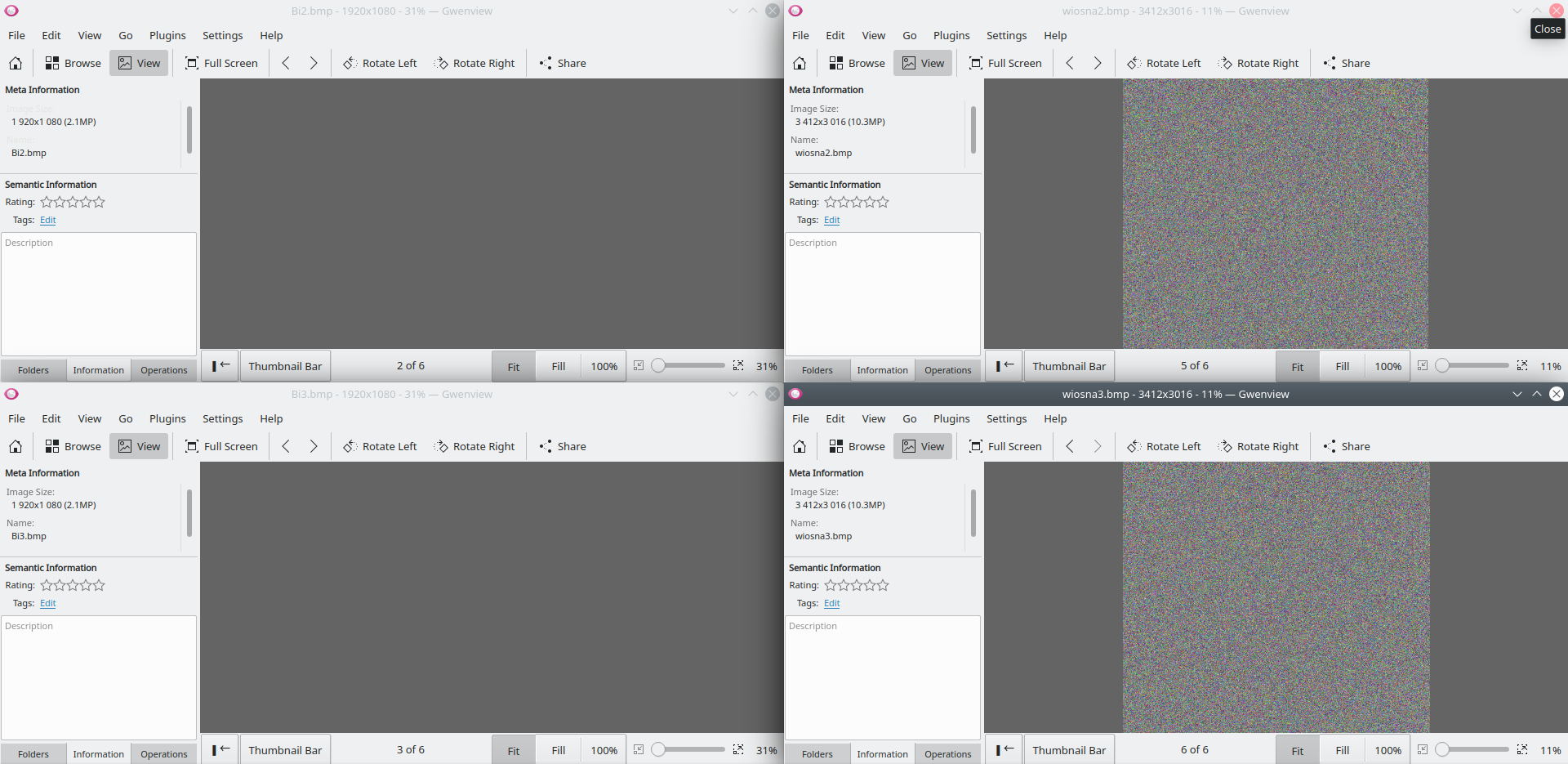
# Zadanie 4.2

## P.4.3

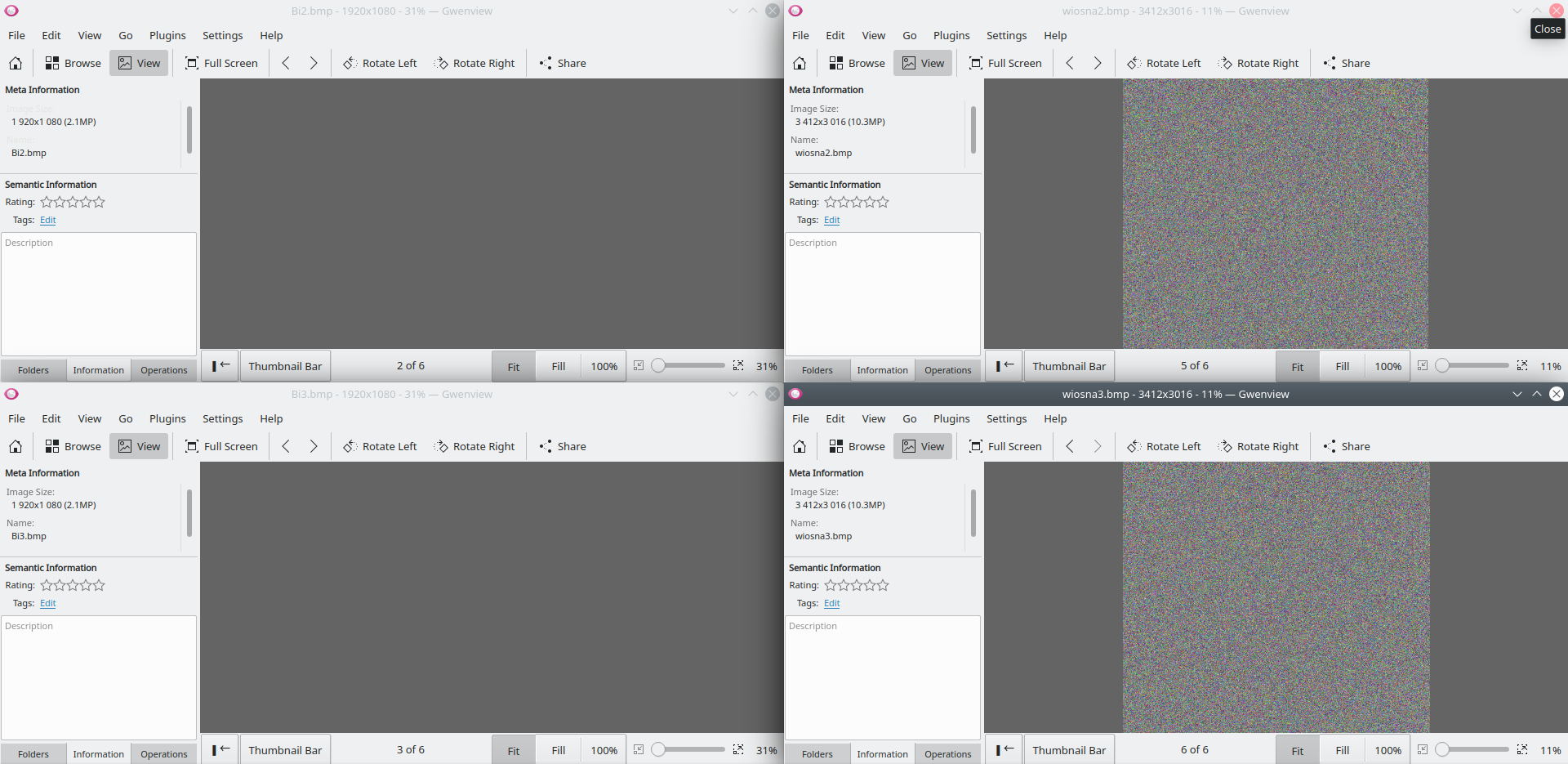
Przebieg szyfrowania obrazów



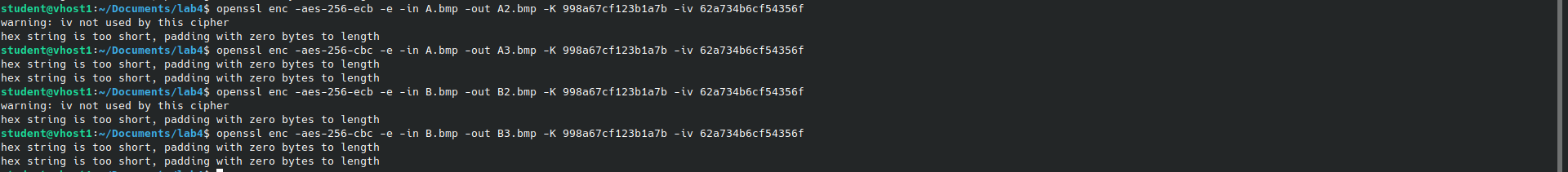
Podgląd obrazów BI.bmp (u góry – szyfrowanie ecb, u dołu – szyfrowanie cbc)



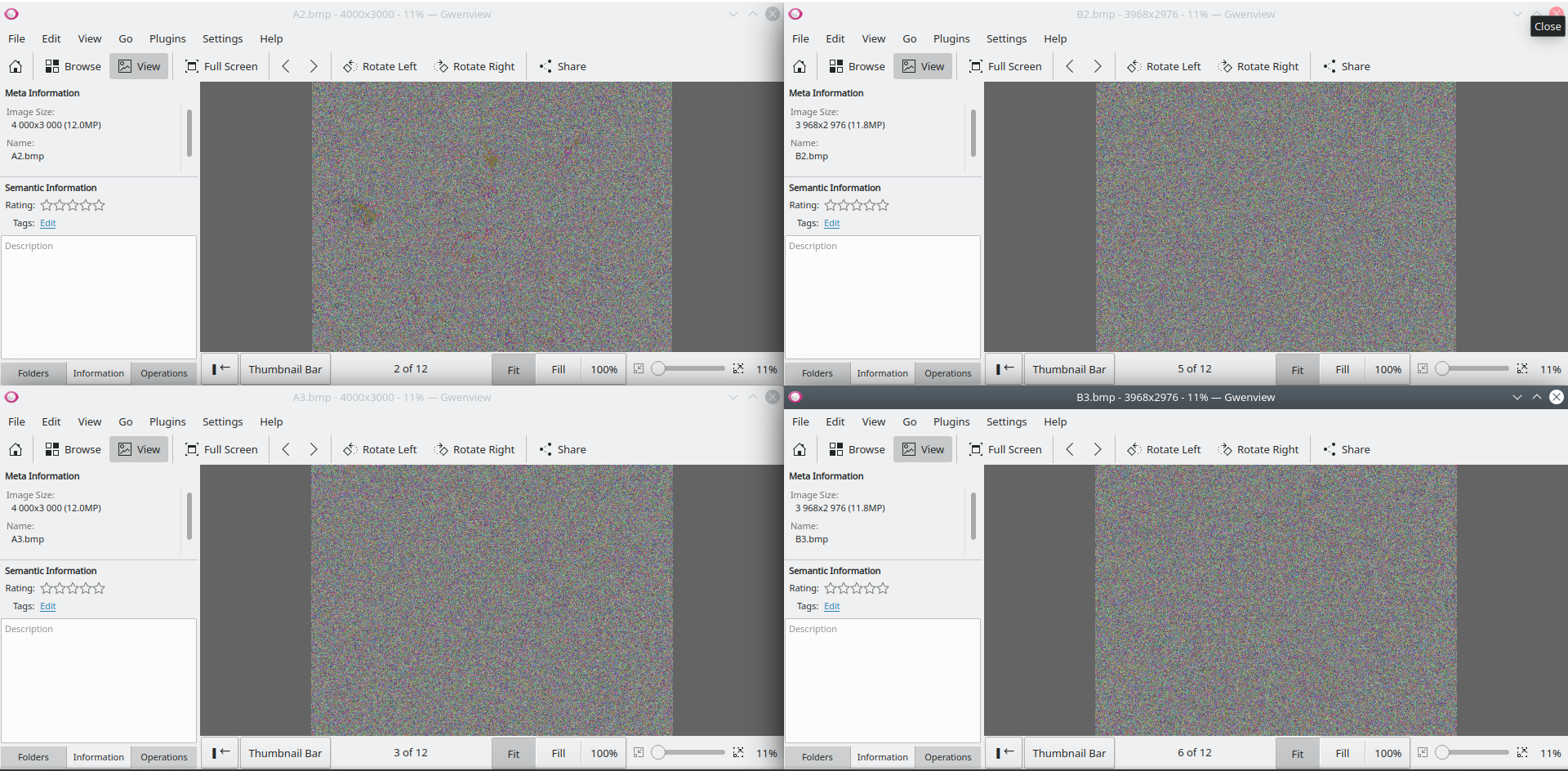
Podgląd obrazów wiosna.bmp (u góry – szyfrowanie ecb, u dołu – szyfrowanie cbc)



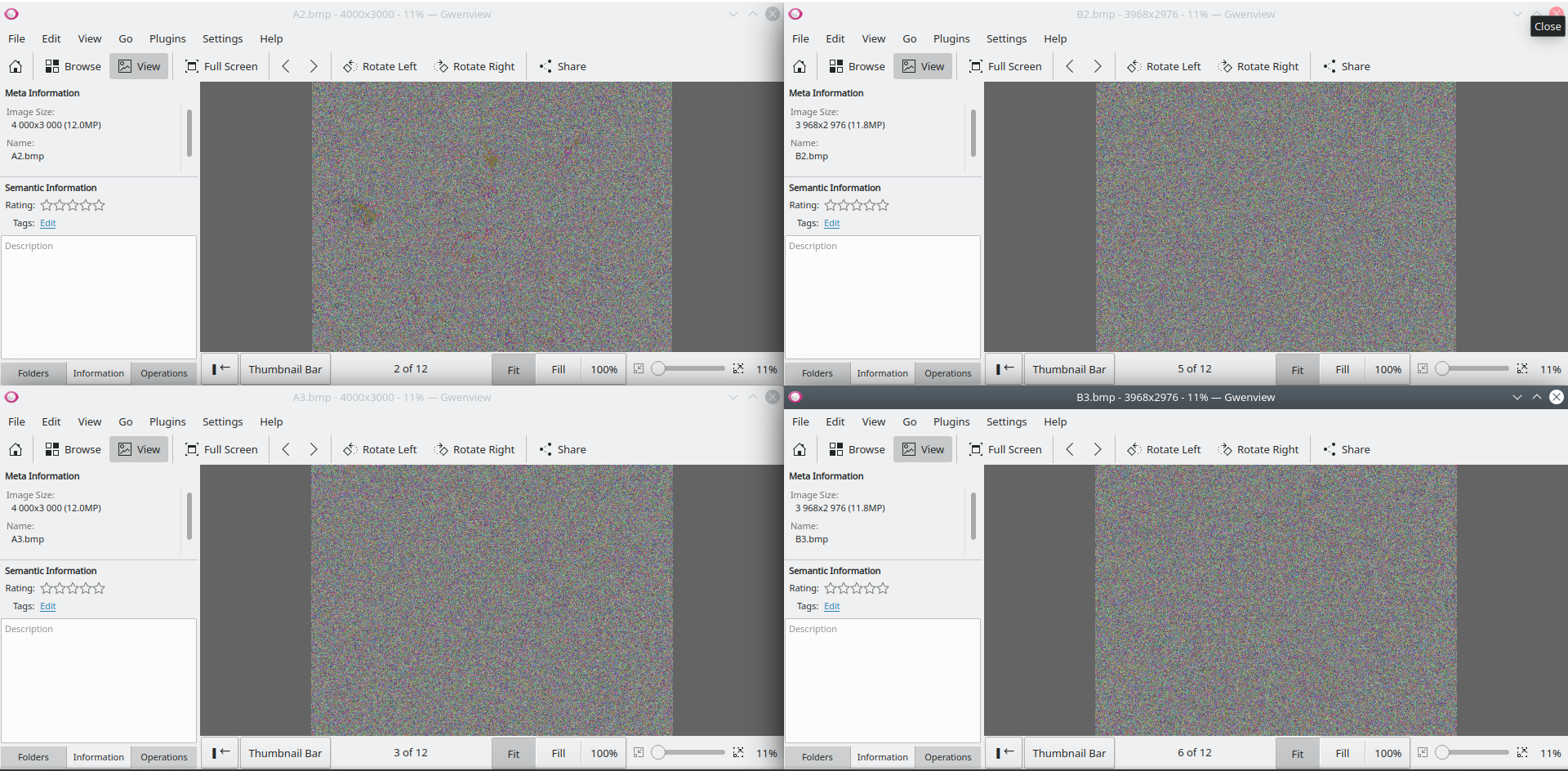
Przebieg szyfrowania obrazów



Podgląd obrazów A.bmp (u góry – szyfrowanie ecb, u dołu – szyfrowanie cbc)



Podgląd obrazów B.bmp (u góry – szyfrowanie ecb, u dołu – szyfrowanie cbc)



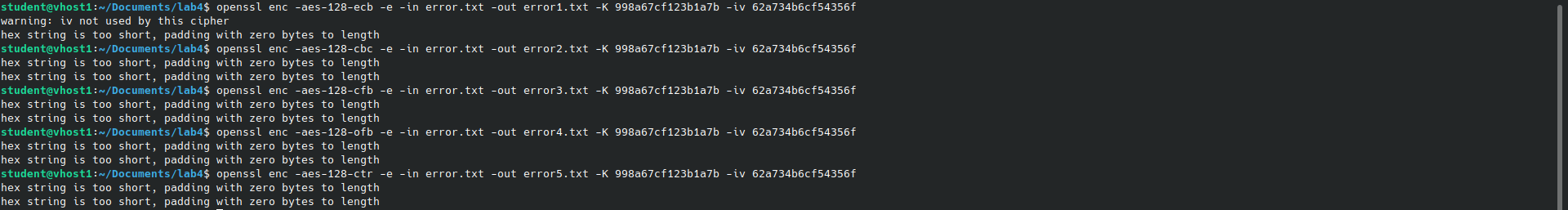
Jeżeli zdjęcie źródłowe jest w przybliżeniu jednego koloru to zaszyfrowany odpowiednik będzie jednolity kolorem, jeżeli zdjęcie jest wielokolorowe i posiada wiele elementów to odpowiednik będzie przedstawiał „losowo” umieszczone piksele w różnych kolorach.

Szyfrogramy utworzone w trybie ECB i CBC są podobne dla danego obrazka.

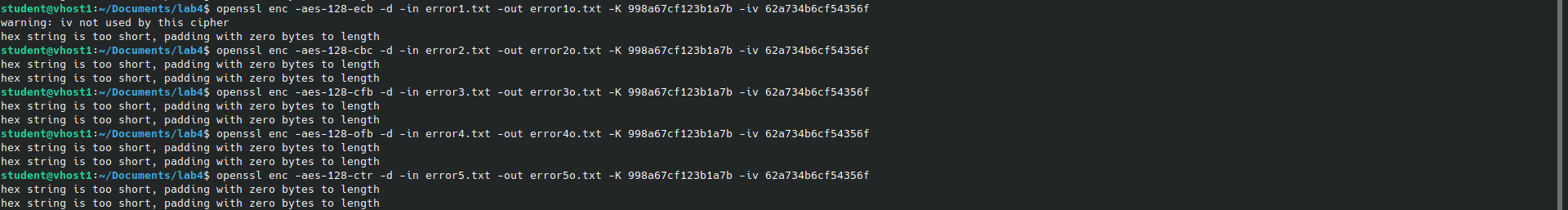
# Zadanie 4.3

## P.4.4

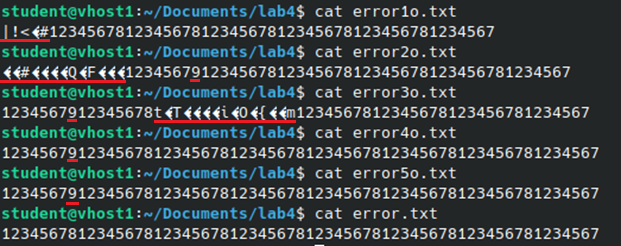
Przebieg szyfrowania



Przebieg odszyfrowywania



Pliki po odszyfrowaniu oraz plik źródłowy



Używałem algorytmu AES-128.

Podmieniłem za pomocą edytora 8 bit w 8 bajcie, widać to również po odszyfrowaniu plików przy pomocy większości trybów, gdzie zakłamanie występuje w 8 znaku, który jest kodowany przez 8 bajt.

Jestem w stanie odczytać w zależności o trybu od 86% do 98% danych, gdyż zostało podmienionych od 1 do 9 znaków.

Przy trybie:

1. ECB tracę 8 znaków,
2. CBC - 9 znaków,
3. CFB - 9 znaków,
4. OFB - 1 znak,
5. CTR - 1 znak.

Przy trybach ECB, CBC, CFB uszkodzenie jednego bitu wpływa na więcej niż jeden bajt. Przy trybach OFB, CTR uszkodzenie jednego bitu wpływa tylko na jeden bajt.

Możemy wyciągnąć wniosek, że tryby OFB i CTR są najodporniejsze na uszkodzenia, więc będą dobrym wyborem w systemach podatnych na uszkodzenia.