Тестовые наборы

Группа А

(Дымовое тестирование)

Тест 1

Тестовая ситуация: открытие текстового файла в кодировке ANSI

Исходные данные:

Текст = «Шифрование»

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ш | и | ф | р | о |
| **11011000** | **11101000** | **11110100** | **11110000** | **11101110** |
| в | а | н | и | е |
| **11100010** | **11100000** | **11101101** | **11101000** | **11100101** |

Ожидаемый результат: «11011000111010001111010011110000111011101110001011100000111011011110100011100101» в поле для исходного текста.

Полученный результат:

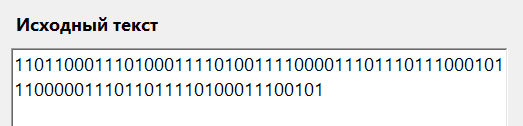


Рисунок 1 – Полученный результат (шифрование). Тест 1

Тест 2

Тестовая ситуация: ввод начального состояния регистра с невалидными символами

Исходные данные:

Исходный текст = 1101100011101000111101001111000011101110111000101110000011101

1011110100011100101

НСР = «11111!abc)))11111 \_=11111^%11111ого11111»

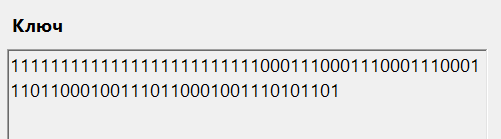
XOR

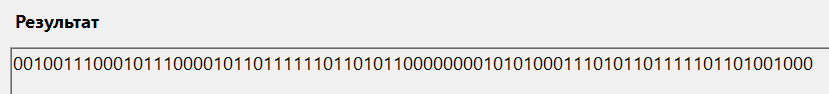
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 11011000 | 11101000 | 11110100 | 11110000 | 11101110 |
| 11111111 | 11111111 | 11111111 | 10001110 | 00111000 |
| 00100111 | 00010111 | 00001011 | 01111110 | 11010110 |
| 11100010 | 11100000 | 11101101 | 11101000 | 11100101 |
| 11100011 | 10110001 | 00111011 | 00010011 | 10101101 |
| 00000001 | 01010001 | 11010110 | 11111011 | 01001000 |

Ожидаемый результат: «11111111111111111111111110001110001110001110001110110001001110110001001110101101» в поле для ключа

«00100111000101110000101101111110110101100000000101010001110101101111101101001000» в поле для результата

Полученный результат:





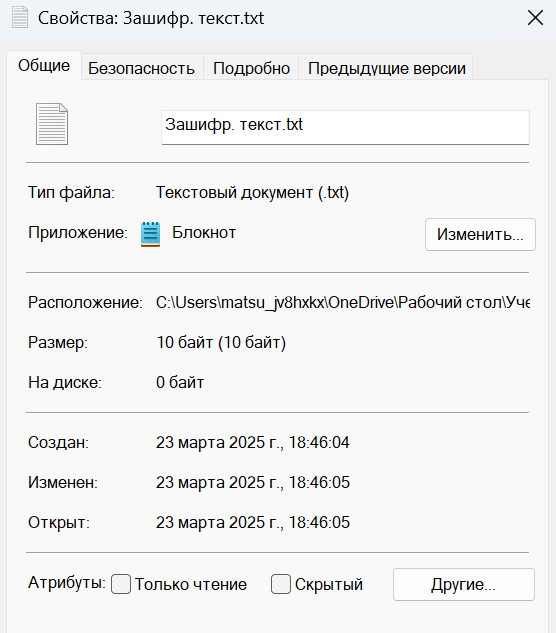
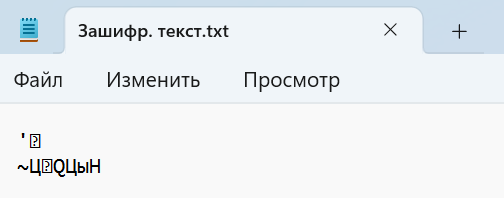
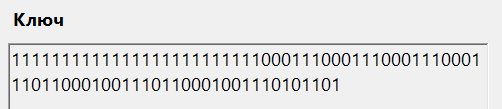
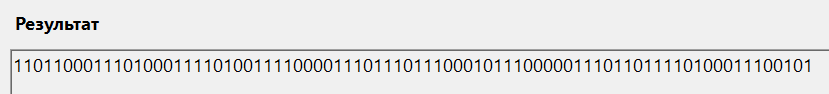


Рисунок 2 – Полученный результат (шифрование). Тест 2





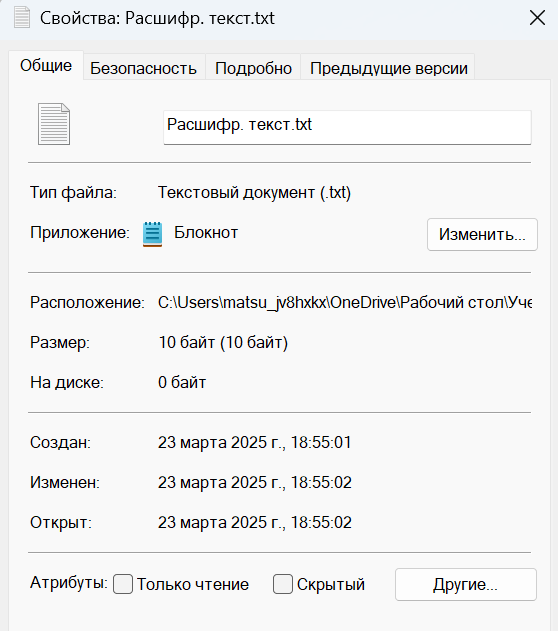
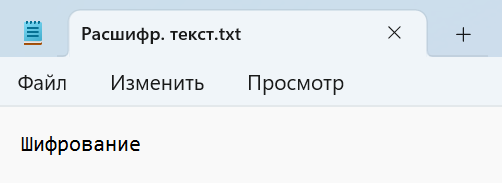


Рисунок 3 – Полученный результат (дешифрование). Тест 2

Тест 3

Тестовая ситуация: шифрование/дешифрование изображения

Исходные данные:



Изображение в формате jpg

Исходный текст (для первых 10 байтов) =

1111111111011000111111111101101100000000100001000000000000001

0000000011000000110

НСР = «11111… \* 99911111ш00000!11111;11111»

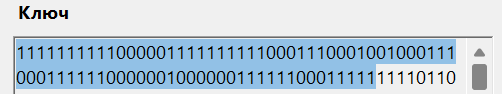
**XOR**

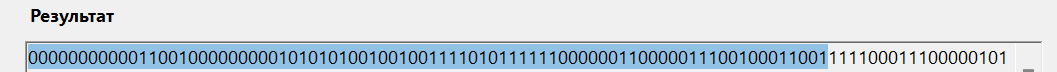
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 11111111 | 11011000 | 11111111 | 11011011 | 00000000 |
| 11111111 | 11000001 | 11111111 | 10001110 | 00100100 |
| 00000000 | 00011001 | 00000000 | 01010101 | 00100100 |
| 10000100 | 00000000 | 00001000 | 00000110 | 00000110 |
| 01110001 | 11111000 | 00010000 | 00111111 | 00011111 |
| 11110101 | 11111000 | 00011000 | 00111001 | 00011001 |

Ожидаемый результат: «11111111110000011111111110001110001001000111000111111000000100000011111100011111» на первых 80 позициях поля для ключа

«00000000000110010000000001010101001001001111010111111000000110000011100100011001» на первых 80 позициях поля для результата

Полученный результат:





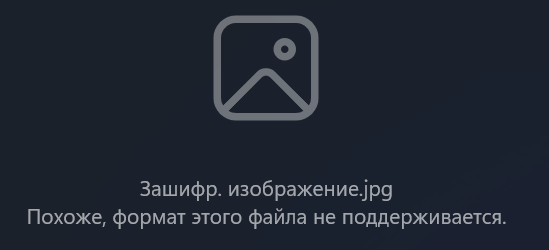
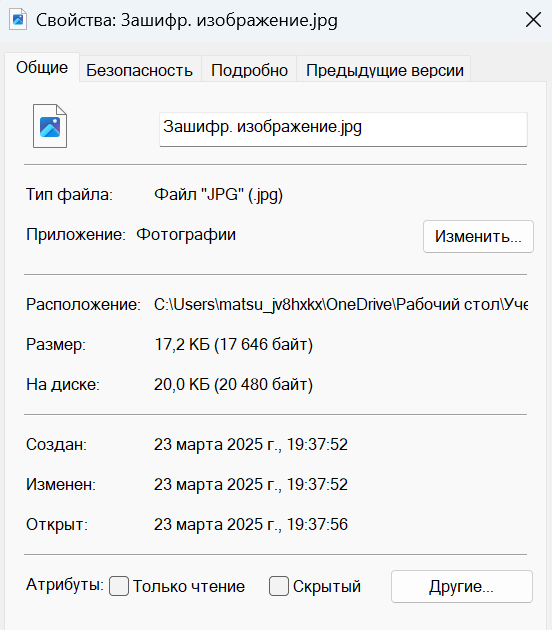
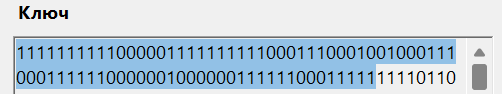
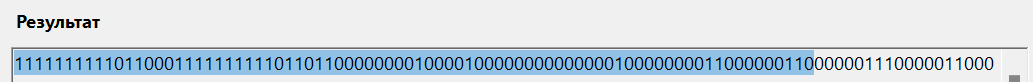
 

Рисунок 4 – Полученный результат (шифрование). Тест 3





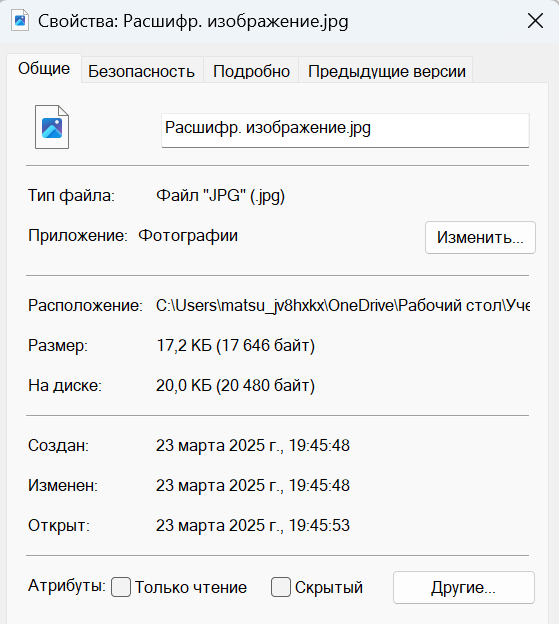


Рисунок 5 – Полученный результат (дешифрование). Тест 3

Тест 4

Тестовая ситуация: шифрование/дешифрование музыки

Исходные данные:

Музыка в формате mp3

Исходный текст (для первых 10 байтов) =

01001001010001000011001100000011000000000000000000000000000000000000000000100011

НСР = «11111!abc)))11111 \_=11111^%11111ого11111»

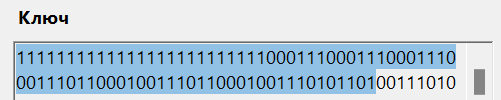
**XOR**

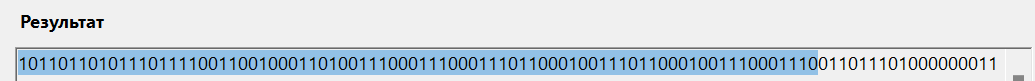
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 01001001 | 01000100 | 00110011 | 00000011 | 00000000 |
| 11111111 | 11111111 | 11111111 | 10001110 | 00111000 |
| 10110110 | 10111011 | 11001100 | 10001101 | 00111000 |
| 00000000 | 00000000 | 00000000 | 00000000 | 00100011 |
| 11100011 | 10110001 | 00111011 | 00010011 | 10101101 |
| 11100011 | 10110001 | 00111011 | 00010011 | 10001110 |

Ожидаемый результат: «11111111111111111111111110001110001110001110001110110001001110110001001110101101» на первых 80 позициях поля для ключа

«10110110101110111100110010001101001110001110001110110001001110110001001110001110» на первых 80 позициях поля для результата

Полученный результат:





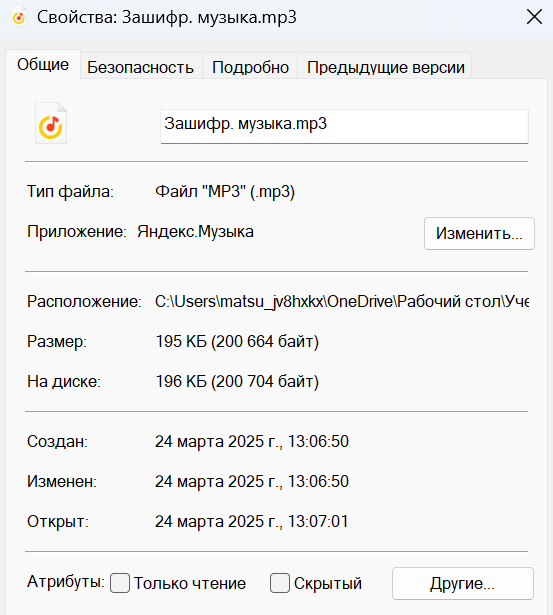
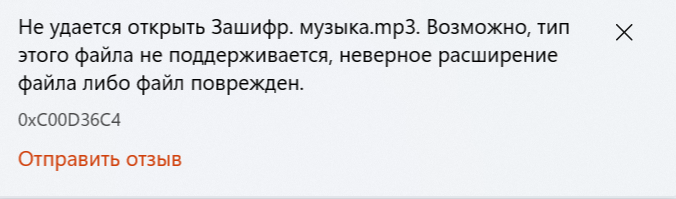
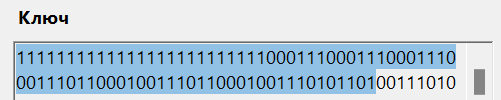
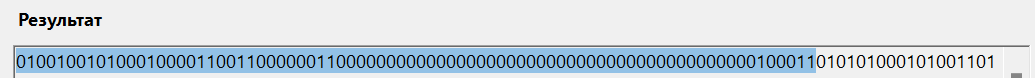


Рисунок 6 – Полученный результат (шифрование). Тест 4





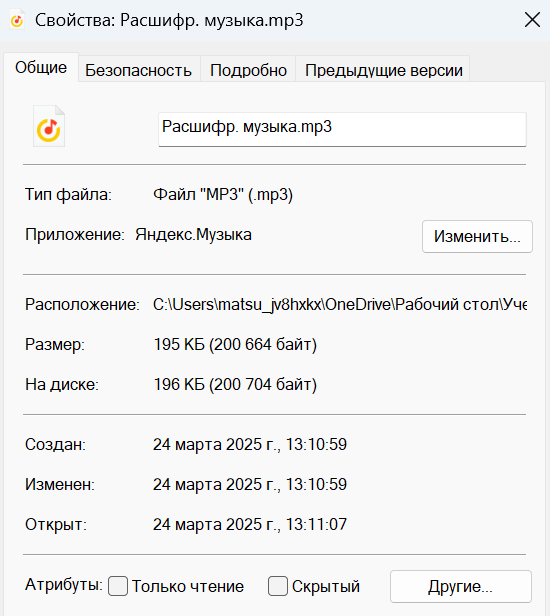
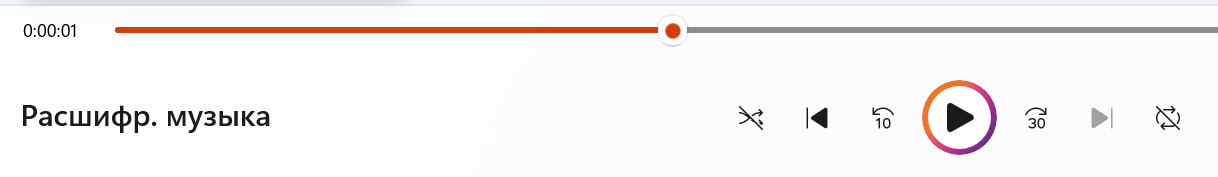
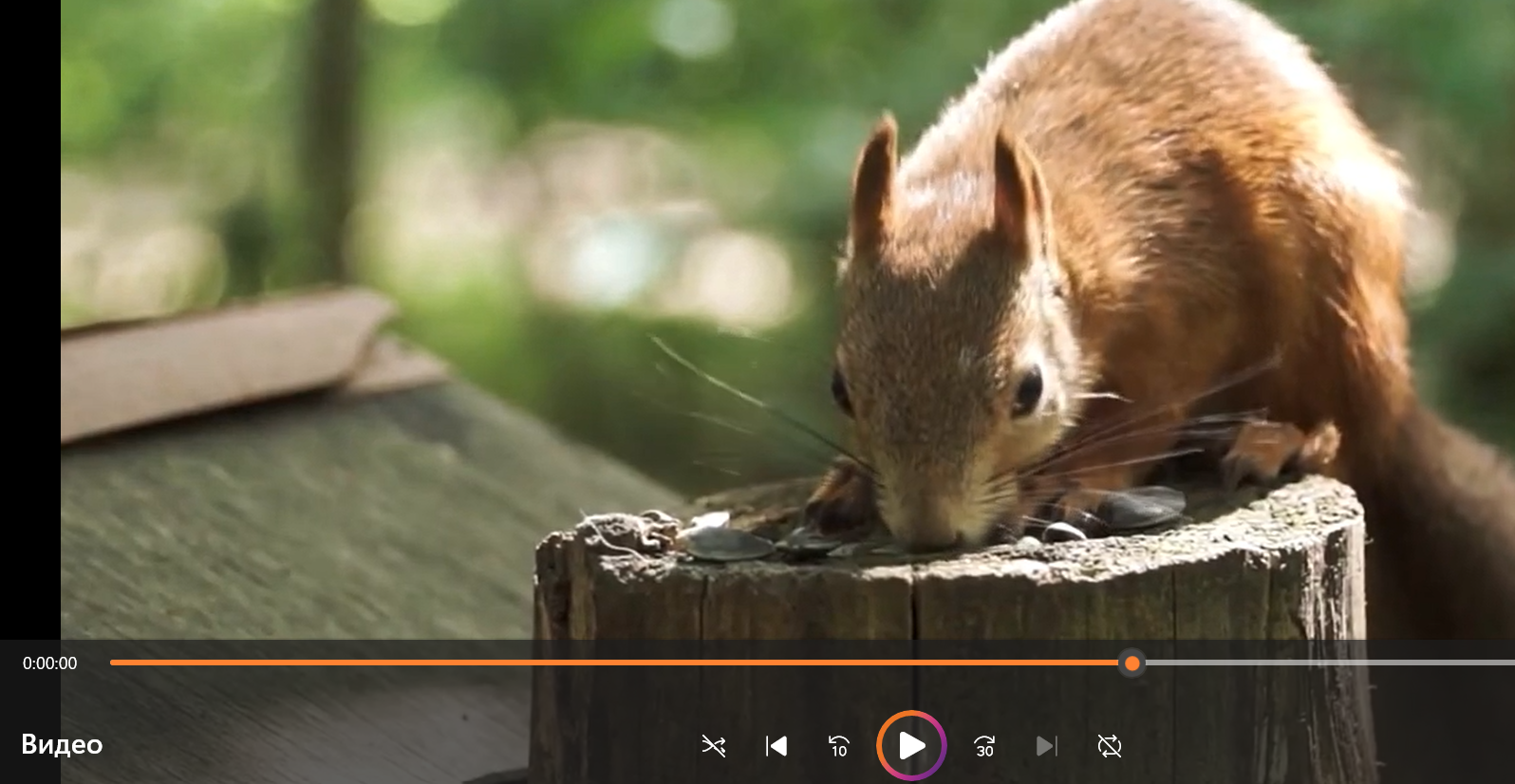


Рисунок 7 – Полученный результат (дешифрование). Тест 4

Тест 5

Тестовая ситуация: шифрование/дешифрование видео

Исходные данные:



Видео в формате mp4

Исходный текст (для первых 10 байтов) =

00000000000000000000000000100000011001100111010001111001011100000110100101110011

НСР = «11111… \* 99911111ш00000!11111;11111»

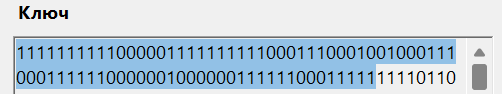
**XOR**

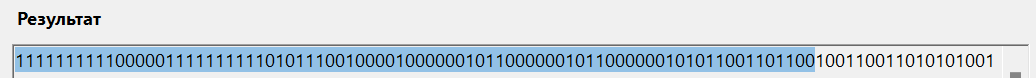
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 00000000 | 00000000 | 00000000 | 00100000 | 01100110 |
| 11111111 | 11000001 | 11111111 | 10001110 | 00100100 |
| 11111111 | 11000001 | 11111111 | 10101110 | 01000010 |
| 01110100 | 01111001 | 01110000 | 01101001 | 01110011 |
| 01110001 | 11111000 | 00010000 | 00111111 | 00011111 |
| 00000101 | 10000001 | 01100000 | 01010110 | 01101100 |

Ожидаемый результат: «11111111110000011111111110001110001001000111000111111000000100000011111100011111» на первых 80 позициях поля для ключа

«11111111110000011111111110101110010000100000010110000001011000000101011001101100» на первых 80 позициях поля для результата

Полученный результат:





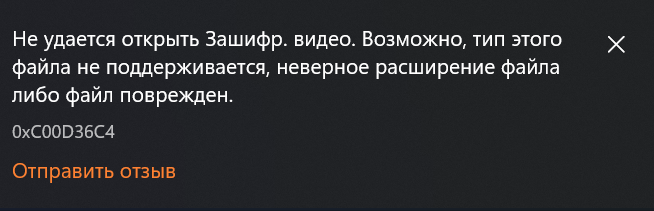
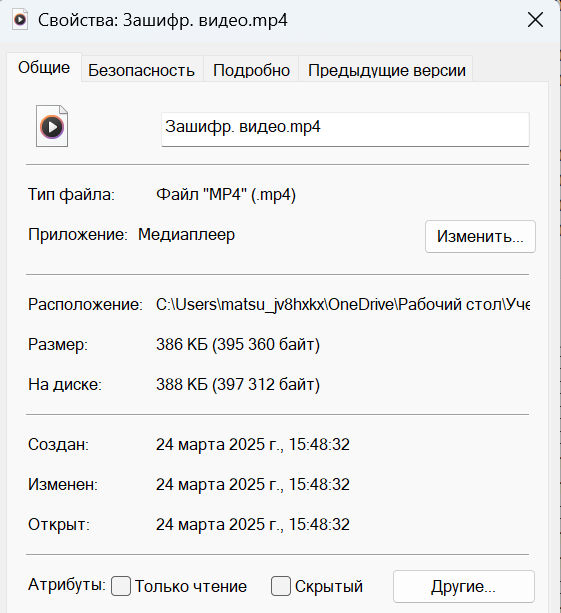
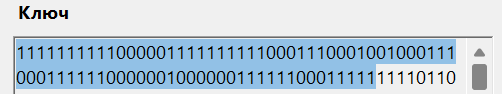
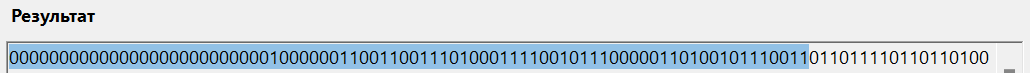
 

Рисунок 8 – Полученный результат (шифрование). Тест 5





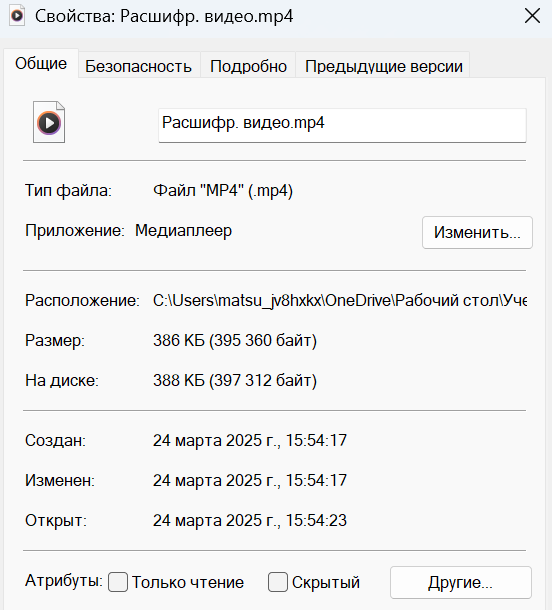
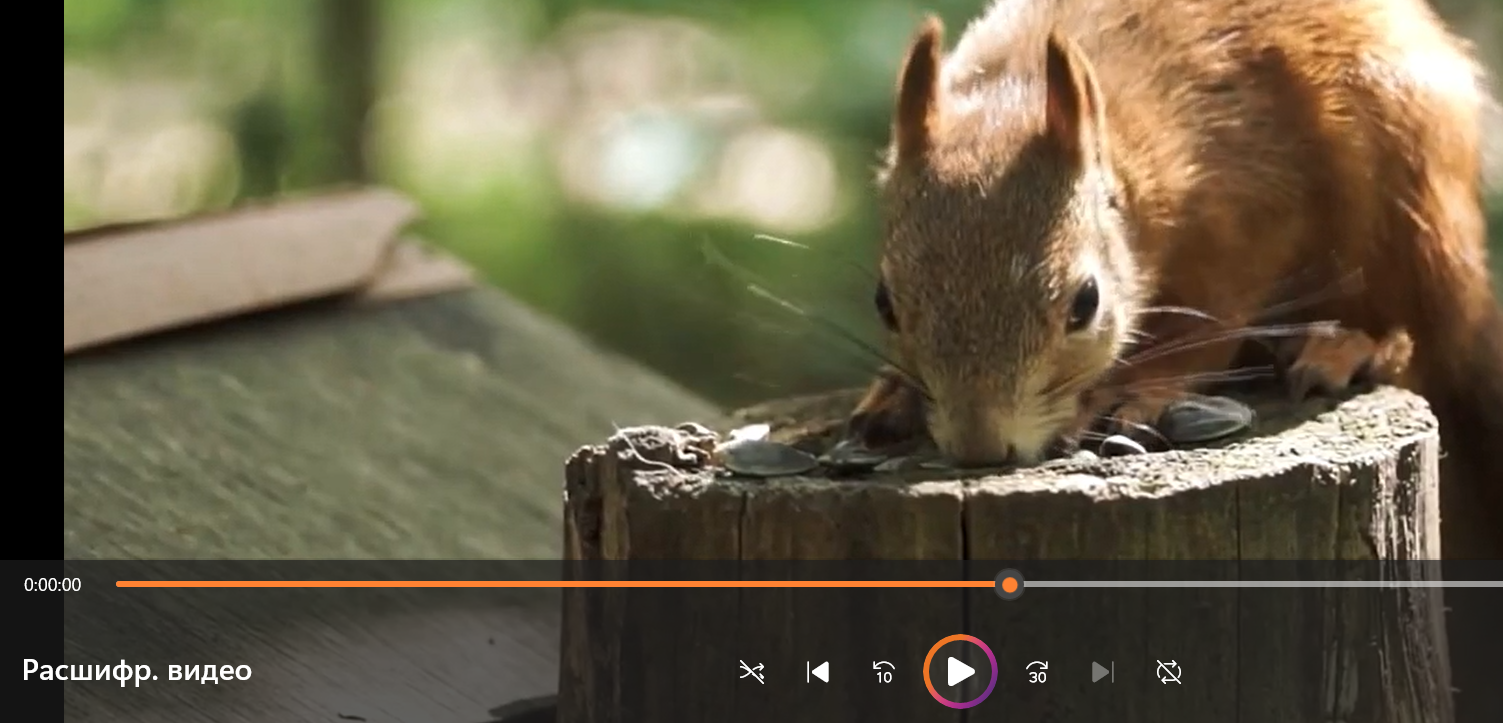


Рисунок 9 – Полученный результат (дешифрование). Тест 5

Группа Б

(Ломание на валидных данных)

Тест 6

Тестовая ситуация: ввод всех нулей в поле для начального состояния ключа

Исходные данные:

Текст = «Шифрование»

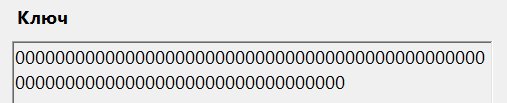
НСР = 0000000000000000000000000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ш | и | ф | р | о |
| **11011000** | **11101000** | **11110100** | **11110000** | **11101110** |
| в | а | н | и | е |
| **11100010** | **11100000** | **11101101** | **11101000** | **11100101** |

Ожидаемый результат: «00000000000000000000000000000000000000000000000000000000000000000000000000000000» в поле для ключа.

«11011000111010001111010011110000111011101110001011100000111011011110100011100101» в поле для результата.

Полученный результат:



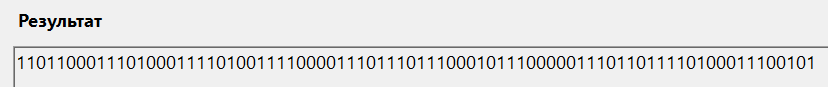
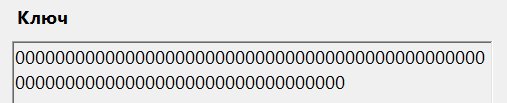


Рисунок 10 – Полученный результат (шифрование). Тест 6



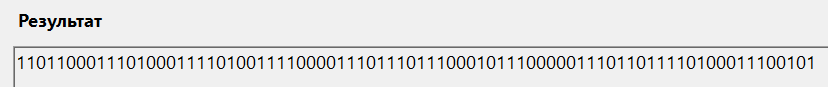


Рисунок 11 – Полученный результат (дешифрование). Тест 6

Тест 7

Тестовая ситуация: ключ длиной более 25

Исходные данные:

Текст = «Шифрование»

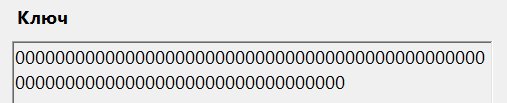
НСР = 000000000000000000000000011111

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ш | и | ф | р | о |
| **11011000** | **11101000** | **11110100** | **11110000** | **11101110** |
| в | а | н | и | е |
| **11100010** | **11100000** | **11101101** | **11101000** | **11100101** |

Ожидаемый результат: «00000000000000000000000000000000000000000000000000000000000000000000000000000000» в поле для ключа.

«11011000111010001111010011110000111011101110001011100000111011011110100011100101» в поле для результата.

Полученный результат:



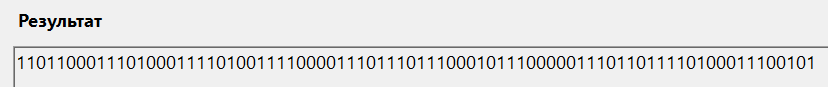
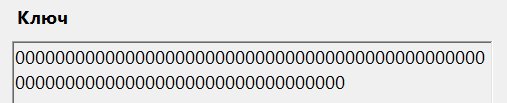


Рисунок 12 – Полученный результат (шифрование). Тест 7



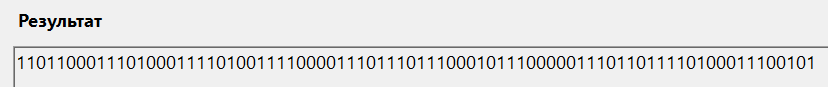


Рисунок 13 – Полученный результат (дешифрование). Тест 7

Тест 8

Тестовая ситуация: исходный текст длиной менее 25

Исходные данные:

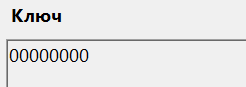
Исходный текст = «11111111»

НСР = 0000000000000000000000000

Ожидаемый результат: «00000000» в поле для ключа.

«11111111» в поле для результата.

Полученный результат:



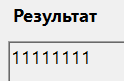
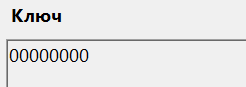


Рисунок 14 – Полученный результат (шифрование). Тест 8



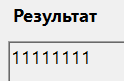


Рисунок 15 – Полученный результат (дешифрование). Тест 8

Группа В

(Ломание на не валидных данных)

Тест 9

Тестовая ситуация: ключ длиной менее 25

Исходные данные:

Текст = «11111111»

Ключ = «111»

Ожидаемый результат: вывод сообщения об ошибке ввода

Полученный результат:

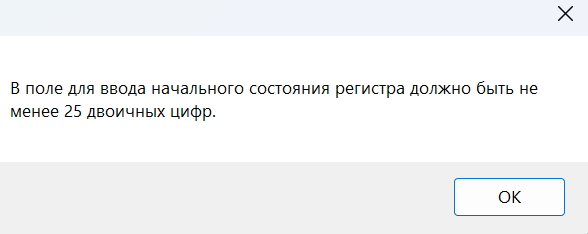


Рисунок 16 – Полученный результат (шифрование). Тест 9

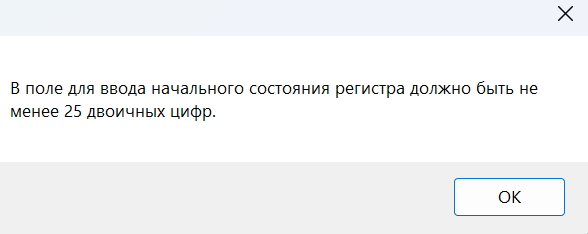


Рисунок 17 – Полученный результат (дешифрование). Тест 9