Лабораторная работа № 3

Тема: исследование гаммирования при шифровании.

Цель: изучить теоретические основы генерации псевдослучайных последовательностей и гаммирования при шифровании данных. Составить программу, позволяющую закодировать текст и выполнить обратное преобразование.

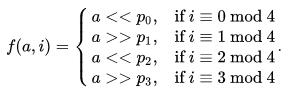
1 Теоретический вопрос

ISAAC (Indirection, Shift, Accumulate, Add and Count) – алгоритм псевдослучайных чисел, принцип работы которого труднее запомнить, чем принципы работы IA и IBAA, зато он имеет по сравнению с ними целый ряд преимуществ. При проектировании ISAAC к нему был предъявлен следующий список требований:

* криптографическая стойкость;
* невозможность получения внутреннего состояния по имеющимся выходным результатам;
* отсутствие коротких циклов;
* отсутствие каких-либо тенденций в распределении бит на всем цикле;
* упорядоченные состояния должны быстро становиться хаотичными.

В отличие от большинства генераторов псевдослучайных чисел, в основе работы которых лежат потоковые шифры, ISAAC разработан без использования линейных регистров сдвига с обратной связью.

Так же, как и в предыдущих алгоритмах, в ISAAC есть массив S, определяющий внутреннее состояние системы, так же состоящий из случайно расположенных в массиве 2n элементов от 0 до 2n-1 длины K бит, итератор i и три переменные a, b и c, отвечающие за предыдущие состояния генератора, массив выходных данных z той же длины, что и S. Однако помимо этих переменных здесь вводятся также переменные p0, p1, p2, p3, которые определяют значение функции, зависящей от обоих итераторов:



Обычно используются p0 = 13; p1 = 6; p2 = 2; p3 = 16.

2 Выполнение шифрования и дешифрования методом сложения по модулю N

В следующих таблицах продемонстрирован процесс шифрования с N = 44

Таблица 1 – Шифрование фразы 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T** | Х | А | С | А | Н | \_ | 2 | 5 | 0 | 0 | 0 | \_ | Т | О | Н | Н |
| **G** | А | Л | Е | К | С | Е | Й | А | Л | Е | К | С | Е | Й | А | Л |
| **T** | 23 | 1 | 19 | 1 | 15 | 34 | 36 | 39 | 44 | 44 | 44 | 34 | 20 | 16 | 15 | 15 |
| **G** | 1 | 13 | 6 | 12 | 19 | 6 | 11 | 1 | 13 | 6 | 12 | 19 | 6 | 11 | 1 | 13 |
| **T+G** | 24 | 14 | 25 | 13 | 34 | 40 | 47 | 40 | 57 | 50 | 56 | 53 | 26 | 27 | 16 | 28 |
| **mod N** | 24 | 14 | 25 | 13 | 34 | 40 | 3 | 40 | 13 | 6 | 12 | 9 | 26 | 27 | 16 | 28 |
| **C** | Ц | М | Ч | Л | \_ | 6 | В | 6 | Л | Е | К | 3 | Ш | Щ | О | Ъ |

Таблица 2 – Шифрование фразы 2

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T** | Х | А | Л | К | И | Н | Г | О | Л | \_ | 1 | 9 | 3 | 9 | \_ | Г |
| **G** | Е | М | Е | Л | Ь | Я | Н | Е | М | Е | Л | Ь | Я | Н | Е | М |
| **T** | 23 | 1 | 13 | 12 | 10 | 15 | 4 | 16 | 13 | 34 | 35 | 43 | 37 | 43 | 34 | 4 |
| **G** | 6 | 14 | 6 | 13 | 30 | 33 | 15 | 6 | 14 | 6 | 13 | 30 | 33 | 15 | 6 | 14 |
| **T+G** | 29 | 15 | 19 | 25 | 40 | 48 | 19 | 22 | 27 | 40 | 48 | 73 | 70 | 58 | 40 | 18 |
| **mod N** | 29 | 15 | 19 | 25 | 40 | 4 | 19 | 22 | 27 | 40 | 4 | 29 | 26 | 14 | 40 | 18 |
| **C** | Ы | Н | С | Ч | 6 | Г | С | Ф | Щ | 6 | Г | Ы | Ш | М | 6 | Р |

Таблица 3 – Шифрование фразы 3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T** | Х | А | Б | А | Р | О | В | С | К | \_ | 6 | 8 | 0 | 0 | 6 | 3 |
| **G** | Т | И | М | О | Ф | Е | Й | Т | И | М | О | Ф | Е | Й | Т | И |
| **T** | 23 | 1 | 2 | 1 | 18 | 16 | 3 | 19 | 12 | 34 | 40 | 42 | 44 | 44 | 40 | 37 |
| **G** | 20 | 10 | 14 | 16 | 22 | 6 | 11 | 20 | 10 | 14 | 16 | 22 | 6 | 11 | 20 | 10 |
| **T+G** | 43 | 11 | 16 | 17 | 40 | 22 | 14 | 39 | 22 | 48 | 56 | 64 | 50 | 55 | 60 | 47 |
| **mod N** | 43 | 11 | 16 | 17 | 40 | 22 | 14 | 39 | 22 | 4 | 12 | 20 | 6 | 11 | 16 | 3 |
| **C** | 9 | Й | О | П | 6 | Ф | М | 5 | Ф | Г | К | Т | Е | Й | О | В |

Таблица 4 – Шифрование фразы 4

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T** | Н | Е | Р | Ю | Н | Г | Р | И | \_ | 2 | 3 | 0 | \_ | К | М |
| **G** | М | А | Р | Ь | Я | Н | А | М | А | Р | Ь | Я | Н | А | М |
| **T** | 15 | 6 | 18 | 32 | 15 | 4 | 18 | 10 | 34 | 36 | 37 | 44 | 34 | 12 | 14 |
| **G** | 14 | 1 | 18 | 30 | 33 | 15 | 1 | 14 | 1 | 18 | 30 | 33 | 15 | 1 | 14 |
| **T+G** | 29 | 7 | 36 | 62 | 48 | 19 | 19 | 24 | 35 | 54 | 67 | 77 | 49 | 13 | 28 |
| **mod N** | 29 | 7 | 36 | 18 | 4 | 19 | 19 | 24 | 35 | 10 | 23 | 33 | 5 | 13 | 28 |
| **C** | Ы | Ё | 2 | Р | Г | С | С | Ц | 1 | И | Х | Я | Д | Л | Ъ |

Таблица 5 – Шифрование фразы 5

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T** | А | М | У | Р | З | Е | Т | \_ | 3 | 5 | 0 | 0 | 0 | \_ | Ш | Т |
| **G** | В | А | Р | В | А | Р | А | В | А | Р | В | А | Р | А | В | А |
| **T** | 1 | 14 | 21 | 18 | 9 | 6 | 20 | 34 | 37 | 39 | 44 | 44 | 44 | 34 | 26 | 20 |
| **G** | 3 | 1 | 18 | 3 | 1 | 18 | 1 | 3 | 1 | 18 | 3 | 1 | 18 | 1 | 3 | 1 |
| **T+G** | 3 | 15 | 39 | 21 | 10 | 24 | 21 | 37 | 38 | 57 | 47 | 45 | 62 | 35 | 29 | 21 |
| **mod N** | 3 | 15 | 39 | 21 | 10 | 24 | 21 | 37 | 38 | 13 | 3 | 1 | 18 | 35 | 29 | 21 |
| **C** | В | Н | 5 | У | И | Ц | У | 3 | 4 | Л | В | А | Р | 1 | Ы | У |

Таблица 6 – Шифрование фразы 6

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **T** | З | Е | Й | С | К | А | Я | \_ | 2 | 5 | 0 | 0 | \_ | Т | О | Н | Н |
| **G** | В | И | Т | А | Л | И | Й | В | И | Т | А | Л | И | Й | В | И | Т |
| **T** | 9 | 6 | 11 | 19 | 12 | 1 | 33 | 34 | 36 | 39 | 44 | 44 | 34 | 20 | 16 | 15 | 15 |
| **G** | 3 | 10 | 20 | 1 | 13 | 10 | 11 | 3 | 10 | 20 | 1 | 13 | 10 | 11 | 3 | 10 | 20 |
| **T+G** | 12 | 16 | 31 | 20 | 25 | 11 | 44 | 37 | 46 | 59 | 45 | 57 | 44 | 31 | 19 | 25 | 35 |
| **mod N** | 12 | 16 | 31 | 20 | 25 | 11 | 44 | 37 | 2 | 15 | 1 | 13 | 44 | 31 | 19 | 25 | 35 |
| **C** | К | О | Э | Т | Ч | Й | 0 | 3 | Б | Н | А | Л | 0 | Э | С | Ч | 1 |

В следующих таблицах продемонстрирован процесс дешифрования с N = 44

Таблица 7 – Дешифрование фразы 7

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C** | Й | Ё | Ё | Л | Щ | Д | 2 | 8 | З | Б | Д | Е | Я | Ш | С |
| **G** | Б | Е | Р | И | Н | Г | Б | Е | Р | И | Н | Г | Б | Е | Р |
| **C** | 11 | 7 | 7 | 13 | 27 | 5 | 36 | 42 | 9 | 2 | 5 | 6 | 33 | 26 | 19 |
| **G** | 2 | 6 | 18 | 10 | 15 | 4 | 2 | 6 | 18 | 10 | 15 | 4 | 2 | 6 | 18 |
| **C-G** | 9 | 1 | -11 | 3 | 12 | 1 | 34 | 36 | -9 | -8 | -10 | 2 | 31 | 20 | 1 |
| **+44** | 53 | 45 | 33 | 47 | 56 | 45 | 78 | 80 | 35 | 36 | 34 | 46 | 75 | 64 | 45 |
| **modN** | 9 | 1 | 33 | 3 | 12 | 1 | 34 | 36 | 35 | 36 | 34 | 2 | 31 | 20 | 1 |
| **О.Т.** | З | А | Я | В | К | А | \_ | 2 | 1 | 2 | \_ | Б | Э | Т | А |

Таблица 8 – Дешифрование фразы 8

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C** | 8 | О | Я | М | \_ | Д | Н | Н | Р | 3 | А | Н | Ф | А | Е | 5 |
| **G** | Х | И | Н | Г | А | Н | Х | И | Н | Г | А | Н | Х | И | Н | Г |
| **C** | 42 | 16 | 33 | 14 | 34 | 5 | 15 | 15 | 18 | 37 | 1 | 15 | 22 | 1 | 6 | 39 |
| **G** | 23 | 10 | 15 | 4 | 1 | 15 | 23 | 10 | 15 | 4 | 1 | 15 | 23 | 10 | 15 | 4 |
| **C-G** | 19 | 6 | 18 | 10 | 33 | -10 | -8 | 5 | 3 | 33 | 0 | 0 | -1 | -9 | -9 | 35 |
| **+44** | 63 | 50 | 62 | 54 | 77 | 34 | 36 | 49 | 47 | 77 | 44 | 44 | 43 | 35 | 35 | 79 |
| **modN** | 19 | 6 | 18 | 10 | 33 | 34 | 36 | 5 | 3 | 33 | 0 | 0 | 43 | 35 | 35 | 35 |
| **О.Т.** | С | Е | Р | И | Я | \_ | 2 | Д | В | Я | 0 | 0 | 9 | 1 | 1 | 1 |

Таблица 9 – Дешифрование фразы 9

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C** | И | Г | Я | Ё | Щ | Т | 6 | А | В | \_ | Ю | Г | Й | М | Б |
| **G** | Д | А | Л | Я | Н | Ь | Д | А | Л | Я | Н | Ь | Д | А | Л |
| **C** | 10 | 4 | 33 | 7 | 27 | 20 | 40 | 1 | 3 | 34 | 32 | 4 | 11 | 14 | 2 |
| **G** | 5 | 1 | 13 | 33 | 15 | 30 | 5 | 1 | 13 | 33 | 15 | 30 | 5 | 1 | 13 |
| **C-G** | 5 | 3 | 20 | -26 | 12 | -10 | 35 | 0 | -10 | 1 | 17 | -26 | 6 | 13 | -11 |
| **+44** | 49 | 47 | 64 | 18 | 56 | 34 | 79 | 44 | 34 | 45 | 61 | 18 | 50 | 57 | 33 |
| **modN** | 5 | 3 | 20 | 18 | 12 | 34 | 35 | 0 | 34 | 1 | 17 | 18 | 6 | 13 | 33 |
| **044** | 5 | 3 | 20 | 18 | 12 | 34 | 35 | 44 | 34 | 1 | 17 | 18 | 6 | 13 | 33 |
| **О.Т.** | Д | В | Т | Р | К | \_ | 1 | 0 | \_ | А | П | Р | Е | Л | Я |

Таблица 10 – Дешифрование фразы 10

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C** | Э | Ф | Э | Д | Ц | Х | Ф | Ъ | М | Д | Г | 2 | К | Ё | Г | У |
| **G** | К | О | Л | Ы | М | А | К | О | Л | Ы | М | А | К | О | Л | Ы |
| **C** | 31 | 22 | 31 | 5 | 24 | 23 | 22 | 28 | 14 | 5 | 4 | 36 | 12 | 7 | 4 | 21 |
| **G** | 12 | 16 | 13 | 29 | 14 | 1 | 12 | 16 | 13 | 29 | 14 | 1 | 12 | 16 | 13 | 29 |
| **C-G** | 19 | 6 | 18 | -24 | 10 | 22 | 10 | 12 | 1 | -24 | -10 | 35 | 0 | -9 | -9 | -8 |
| **+44** | 63 | 50 | 62 | 20 | 54 | 66 | 54 | 56 | 45 | 20 | 34 | 79 | 44 | 35 | 35 | 36 |
| **modN** | 19 | 6 | 18 | 20 | 10 | 22 | 10 | 12 | 1 | 20 | 34 | 35 | 0 | 35 | 35 | 36 |
| **044** | 19 | 6 | 18 | 20 | 10 | 22 | 10 | 12 | 1 | 20 | 34 | 35 | 44 | 35 | 35 | 36 |
| **О.Т.** | С | Е | Р | Т | И | Ф | И | К | А | Т | \_ | 1 | 0 | 1 | 1 | 2 |

Таблица 11 – Дешифрование фразы 11

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C** | Я | 6 | Ш | Ю | Т | У | Е | Н | Й | Й | К | Д | Ё | М | О |
| **G** | О | Х | О | Т | С | К | О | Х | О | Т | С | К | О | Х | О |
| **C** | 33 | 40 | 26 | 32 | 20 | 21 | 6 | 15 | 11 | 11 | 12 | 5 | 7 | 14 | 16 |
| **G** | 16 | 23 | 16 | 20 | 19 | 12 | 16 | 23 | 16 | 20 | 19 | 12 | 16 | 23 | 16 |
| **C-G** | 17 | 17 | 10 | 12 | 1 | 9 | -10 | -8 | 5 | -9 | -7 | -7 | -9 | -9 | 0 |
| **44** | 61 | 61 | 54 | 56 | 45 | 53 | 34 | 36 | 49 | 35 | 37 | 37 | 35 | 35 | 44 |
| **modN** | 17 | 17 | 10 | 12 | 1 | 9 | 34 | 36 | 5 | 35 | 37 | 37 | 35 | 35 | 0 |
| **044** | 17 | 17 | 10 | 12 | 1 | 9 | 34 | 36 | 5 | 35 | 37 | 37 | 35 | 35 | 44 |
| **О.Т.** | П | П | И | К | А | З | \_ | 2 | Д | 1 | 3 | 3 | 1 | 1 | 0 |

Таблица 12 – Дешифрование фразы 12

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **C** | С | В | И | Ь | Ё | 6 | Ё | П | Й | Р | 0 | З | 1 | 2 | 2 | М |
| **G** | А | Б | А | К | А | Н | А | Б | А | К | А | Н | А | Б | А | К |
| **C** | 19 | 3 | 10 | 30 | 7 | 40 | 7 | 17 | 11 | 18 | 44 | 9 | 35 | 36 | 36 | 14 |
| **G** | 1 | 2 | 1 | 12 | 1 | 15 | 1 | 2 | 1 | 12 | 1 | 15 | 1 | 2 | 1 | 12 |
| **C-G** | 18 | 1 | 9 | 18 | 6 | 25 | 6 | 15 | 10 | 6 | 43 | -6 | 34 | 34 | 35 | 2 |
| **44** | 62 | 45 | 53 | 62 | 50 | 69 | 50 | 59 | 54 | 50 | 87 | 38 | 78 | 78 | 79 | 46 |
| **modN** | 18 | 1 | 9 | 18 | 6 | 25 | 6 | 15 | 10 | 6 | 43 | 38 | 34 | 34 | 35 | 2 |
| **О.Т.** | Р | А | З | Р | Е | Ч | Е | Н | И | Е | 9 | 4 | \_ | \_ | 1 | Б |

3 Блок схемы

На рисунке 1 представлена блок схема шифрования и дешифрования сообщения методом сложения по модулю N.



Рисунок 1 – Блок схемы шифрования и дешифрования

На рисунке 2 представлена блок схема ГПСП.



Рисунок 2 – Блок схема ГПСП

4 Описание работы программы

На вход принимаются 2 строки Ключ и Текст. Ключ используется при генерации гаммы и может быть любой последовательностью символов в кодировке CP866. Текст в зависимости от направления алгоритма необходимо передавать в кодировке CP866 для шифрования и шестнадцатеричном формате для дешифрования. Сам по себе алгоритм двусторонний, однако необходимость задания разных форм обусловлена наличием в зашифрованной строке непечатных символов, что затрудняет их запись и воспроизведение.

На рисунке 3 представлен скриншот интерфейса разработанной программы.

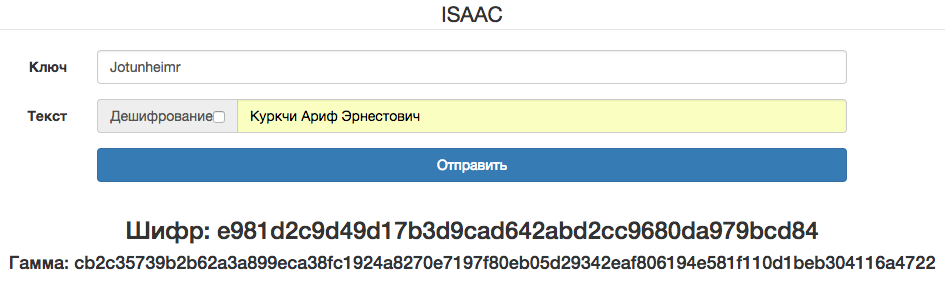


Рисунок 3 – Интерфейс программы

5 Текст программы

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 1 | <?php | | 2 |  | | 3 | *class* ISAAC { | | 4 | private $m, $a, $b, $c; | | 5 | public  $r; | | 6 |  | | 7 | public *function* isaac() { | | 8 | $c = ++$this->c; | | 9 | $b = $this->b += $c; | | 10 | $a = $this->a; | | 11 |  | | 12 | $m =& $this->m; | | 13 | $r = array(); | | 14 |  | | 15 | for ($i = 0; $i < 256; ++$i) { | | 16 | $x = $m[$i]; | | 17 | switch ($i & 3) { | | 18 | case 0: $a ^= ($a << 13); break; | | 19 | case 1: $a ^= ($a >>  6) & 0x03ffffff; break; | | 20 | case 2: $a ^= ($a <<  2); break; | | 21 | case 3: $a ^= ($a >> 16) & 0x0000ffff; break; | | 22 | } | | 23 | $a += $m[$i ^ 128]; $a &= 0xffffffff; | | 24 | $m[$i] = $y = ($m[($x >>  2) & 255] + $a + $b) & 0xffffffff; | | 25 | $r[$i] = $b = ($m[($y >> 10) & 255] + $x) & 0xffffffff; | | 26 | } | | 27 |  | | 28 | $this->a = $a; | | 29 | $this->b = $b; | | 30 | $this->c = $c; | | 31 | $this->r = $r; | | 32 | } | | 33 |  | | 34 | public *function* rand(*$len* = 1) { | | 35 | $result = ''; | | 36 | for($i=0;$i<$len;$i+=2) { | | 37 | if (empty($this->r)) { | | 38 | $this->isaac(); | | 39 | } | | 40 | $result .= dechex(array\_pop($this->r)); | | 41 | } | | 42 | return $result; | | 43 | } | | 44 |  | | 45 | private static *function* mix( &*$a*, &*$b*, &*$c*, &*$d*, &*$e*, &*$f*, &*$g*, &*$h* ) { | | 46 | $a ^= ($b << 11);              $d += $a; $b += $c; | | 47 | $b ^= ($c >>  2) & 0x3fffffff; $e += $b; $c += $d; | | 48 | $c ^= ($d <<  8);              $f += $c; $d += $e; | | 49 | $d ^= ($e >> 16) & 0x0000ffff; $g += $d; $e += $f; | | 50 | $e ^= ($f << 10);              $h += $e; $f += $g; | | 51 | $f ^= ($g >>  4) & 0x0fffffff; $a += $f; $g += $h; | | 52 | $g ^= ($h <<  8);              $b += $g; $h += $a; | | 53 | $h ^= ($a >>  9) & 0x007fffff; $c += $h; $a += $b; | | 54 |  | | 55 | $a &= 0xffffffff; $b &= 0xffffffff; $c &= 0xffffffff; $d &= 0xffffffff; | | 56 | $e &= 0xffffffff; $f &= 0xffffffff; $g &= 0xffffffff; $h &= 0xffffffff; | | 57 | } | | 58 |  | | 59 | public *function* \_\_construct ( *$seed* = null ) { | | 60 | $this->a = $this->b = $this->c = 0; | | 61 | $this->m = array\_fill(0, 256, 0); | | 62 | $m =& $this->m; | | 63 |  | | 64 | $a = $b = $c = $d = $e = $f = $g = $h = 0x9e3779b9; | | 65 |  | | 66 | for ($i = 0; $i < 4; ++$i) { | | 67 | *ISAAC*::mix($a, $b, $c, $d, $e, $f, $g, $h); | | 68 | } | | 69 |  | | 70 | if ( isset($seed) ) { | | 71 | if ( is\_string($seed) ) { | | 72 | $seed = array\_values(unpack("V256", pack("a1024", $seed))); | | 73 | } | | 74 |  | | 75 | for ($i = 0; $i < 256; $i += 8) { | | 76 | $a += $seed[$i  ]; $b += $seed[$i+1]; | | 77 | $c += $seed[$i+2]; $d += $seed[$i+3]; | | 78 | $e += $seed[$i+4]; $f += $seed[$i+5]; | | 79 | $g += $seed[$i+6]; $h += $seed[$i+7]; | | 80 | *ISAAC*::mix($a, $b, $c, $d, $e, $f, $g, $h); | | 81 | $m[$i  ] = $a; $m[$i+1] = $b; $m[$i+2] = $c; $m[$i+3] = $d; | | 82 | $m[$i+4] = $e; $m[$i+5] = $f; $m[$i+6] = $g; $m[$i+7] = $h; | | 83 | } | | 84 |  | | 85 | for ($i = 0; $i < 256; $i += 8) { | | 86 | $a += $m[$i  ]; $b += $m[$i+1]; $c += $m[$i+2]; $d += $m[$i+3]; | | 87 | $e += $m[$i+4]; $f += $m[$i+5]; $g += $m[$i+6]; $h += $m[$i+7]; | | 88 | *ISAAC*::mix($a, $b, $c, $d, $e, $f, $g, $h); | | 89 | $m[$i  ] = $a; $m[$i+1] = $b; $m[$i+2] = $c; $m[$i+3] = $d; | | 90 | $m[$i+4] = $e; $m[$i+5] = $f; $m[$i+6] = $g; $m[$i+7] = $h; | | 91 | } | | 92 | } | | 93 | else { | | 94 | for ($i = 0; $i < 256; $i += 8) { | | 95 | *ISAAC*::mix($a, $b, $c, $d, $e, $f, $g, $h); | | 96 | $m[$i  ] = $a; $m[$i+1] = $b; $m[$i+2] = $c; $m[$i+3] = $d; | | 97 | $m[$i+4] = $e; $m[$i+5] = $f; $m[$i+6] = $g; $m[$i+7] = $h; | | 98 | } | | 99 | } | | 100 |  | | 101 | // fill in the first set of results | | 102 | $this->isaac(); | | 103 | } | | 104 | } | | 105 |  | | 106 | *function* isaac(*$key*, *$len*) { | | 107 | return (new *ISAAC*($key))->rand($len); | | 108 | } | | 109 |  | | 110 | *function* gamma(*$gamma*, *$text*) { | | 111 | $result = ''; | | 112 | for($i=0;$i<strlen($text);$i++) { | | 113 | $result .= dechex(ord($gamma[$i]) ^ ord($text[$i])); | | 114 | } | | 115 | return $result; | | 116 | } | | 117 |  | | 118 | $result = false; | | 119 | $gamma = false; | | 120 | if(isset($\_REQUEST['key']) && isset($\_REQUEST['text'])) { | | 121 | $key = iconv("UTF-8", "CP866", $\_REQUEST['key']); | | 122 | $hex\_mode = isset($\_REQUEST['text\_hex']) && $\_REQUEST['text\_hex'] == '1'; | | 123 | if($hex\_mode) { | | 124 | $text = ''; | | 125 | $tmp = $\_REQUEST['text']; | | 126 | for($i=0;$i<strlen($tmp)-1;$i+=2) { | | 127 | $text .= chr(hexdec($tmp[$i].$tmp[$i+1])); | | 128 | } | | 129 | } else { | | 130 | $text = iconv("UTF-8", "CP866", $\_REQUEST['text']); | | 131 | } | | 132 | $gamma = isaac($key, strlen($text)); | | 133 | $gammed = gamma($gamma, $text); | | 134 | if($hex\_mode) { | | 135 | $result = ''; | | 136 | for($i=0;$i<strlen($gammed)-1;$i+=2) { | | 137 | $result .= chr(hexdec($gammed[$i].$gammed[$i+1])); | | 138 | } | | 139 | } else { | | 140 | $result = $gammed; | | 141 | } | | 142 | $result = iconv("CP866", "UTF-8", $result); | | 143 | } | | 144 | ?> | | 145 |  | | 146 | <html> | | 147 | <head> | | 148 | <title>Gamma ISAAC</title> | | 149 | <link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css"> | | 150 | <link rel="stylesheet" type="text/css" href="assets/css/main.css"> | | 151 | <meta charset="utf-8"> | | 152 | </head> | | 153 | <body> | | 154 | <div class="container"> | | 155 | <form class="form-horizontal text-center"> | | 156 | <fieldset> | | 157 |  | | 158 | <!-- Form Name --> | | 159 | <legend>ISAAC</legend> | | 160 |  | | 161 | <!-- Text input--> | | 162 | <div class="form-group"> | | 163 | <label class="col-md-2 control-label" for="key">Ключ</label> | | 164 | <div class="col-md-8"> | | 165 | <input id="key" name="key" type="text" placeholder="" class="form-control input-md" required=""> | | 166 | </div> | | 167 | </div> | | 168 |  | | 169 | <!-- Prepended checkbox --> | | 170 | <div class="form-group"> | | 171 | <label class="col-md-2 control-label" for="text">Текст</label> | | 172 | <div class="col-md-8"> | | 173 | <div class="input-group"> | | 174 | <label class="input-group-addon"> Декодирование | | 175 | <input type="checkbox" name="text\_hex" value="1"> | | 176 | </label> | | 177 | <input id="text" name="text" class="form-control" type="text" placeholder=""> | | 178 | </div> | | 179 | </div> | | 180 | </div> | | 181 |  | | 182 | <!-- Button --> | | 183 | <div class="form-group"> | | 184 | <div class="col-md-8 col-md-offset-2"> | | 185 | <button type="submit" class="btn btn-primary btn-block">Отправить</button> | | 186 | </div> | | 187 | </div> | | 188 | </fieldset> | | 189 | </form> | | 190 | <div class="text-center"> | | 191 | <?php if($result !== false): ?> | | 192 | <h3>Шифр: <?= $result ?></h3> | | 193 | <h4>Гамма: <?= $gamma ?></h4> | | 194 | <?php endif ?> | | 195 | </div> | | 196 | </div> | | 197 | <script src="assets/js/jquery-3.3.1.min.js"></script> | | 198 | <script src="assets/js/bootstrap.min.js"></script> | | 199 | <script src="assets/js/main.js"></script> | | 200 | </body> | | 201 | </html> | |  |

Вывод

В ходе выполнения лабораторной работы были изучены теоретические основы генерации псевдослучайных последовательностей и гаммирования при шифровании данных. Составлена программа, позволяющая закодировать текст и выполнить обратное преобразование с использованием ГПСП ISAAC.