

# Содержание

|                                                            |           |
|------------------------------------------------------------|-----------|
| Об авторе .....                                            | 20        |
| О технических рецензентах .....                            | 20        |
| <b>ВВЕДЕНИЕ</b>                                            | <b>21</b> |
| Для кого предназначена эта книга .....                     | 22        |
| Структура книги .....                                      | 23        |
| Как пользоваться книгой .....                              | 25        |
| Ввод исходного кода .....                                  | 25        |
| Исправление опечаток .....                                 | 26        |
| Правила оформления листингов, принятые в книге .....       | 26        |
| Ресурсы в Интернете .....                                  | 26        |
| Загрузка и установка Python .....                          | 27        |
| Windows .....                                              | 27        |
| macOS .....                                                | 27        |
| Загрузка файла <i>hyperclip.py</i> .....                   | 28        |
| Запуск IDLE .....                                          | 28        |
| Резюме .....                                               | 29        |
| Ждем ваших отзывов! .....                                  | 30        |
| <b>ГЛАВА 1</b>                                             |           |
| <b>ИНСТРУМЕНТЫ “БУМАЖНОЙ” КРИПТОГРАФИИ</b>                 | <b>31</b> |
| Что такое криптография .....                               | 32        |
| Коды и шифры .....                                         | 32        |
| Шифр Цезаря .....                                          | 34        |
| Шифровальный диск .....                                    | 35        |
| Шифрование сообщения с помощью шифровального диска .....   | 36        |
| Дешифрование сообщения с помощью шифровального диска ..... | 37        |
| Шифрование и дешифрование средствами арифметики .....      | 37        |
| Почему не работает двойное шифрование .....                | 39        |
| Резюме .....                                               | 39        |
| <b>ГЛАВА 2</b>                                             |           |
| <b>ПРОГРАММИРОВАНИЕ В ИНТЕРАКТИВНОЙ ОБОЛОЧКЕ</b>           | <b>41</b> |
| Простые арифметические выражения .....                     | 42        |
| Целые и вещественные числа .....                           | 43        |
| Выражения .....                                            | 43        |
| Порядок операций .....                                     | 44        |

|                                        |    |
|----------------------------------------|----|
| Вычисление выражений.....              | 44 |
| Сохранение значений в переменных ..... | 46 |
| Изменение значений переменных .....    | 48 |
| Имена переменных .....                 | 49 |
| Резюме .....                           | 50 |

### **ГЛАВА 3**

#### **СТРОКОВЫЙ ТИП ДАННЫХ И НАПИСАНИЕ ПРОГРАММ 53**

|                                                                            |    |
|----------------------------------------------------------------------------|----|
| Использование строковых значений для работы с текстом.....                 | 54 |
| Конкатенация строк с помощью оператора + .....                             | 55 |
| Репликация строк с помощью оператора * .....                               | 56 |
| Получение символов строки с помощью индексов .....                         | 57 |
| Вывод значений с помощью функции print () .....                            | 60 |
| Вывод экранированных символов .....                                        | 61 |
| Одинарные и двойные кавычки.....                                           | 63 |
| Написание программ в редакторе файлов IDLE .....                           | 64 |
| Исходный код программы “Hello, world!” .....                               | 64 |
| Проверка правильности исходного кода с помощью<br>онлайн-утилиты Diff..... | 65 |
| Использование IDLE для последующего доступа к программе.....               | 66 |
| Сохранение программы .....                                                 | 66 |
| Выполнение программы .....                                                 | 67 |
| Открытие программы, которую вы сохранили ранее .....                       | 68 |
| Как работает программа “Hello, world!” .....                               | 68 |
| Комментарии .....                                                          | 69 |
| Вывод сообщений для пользователя .....                                     | 69 |
| Ввод данных пользователем .....                                            | 70 |
| Завершение программы .....                                                 | 70 |
| Резюме .....                                                               | 71 |

### **ГЛАВА 4**

#### **ОБРАТНЫЙ ШИФР 73**

|                                               |    |
|-----------------------------------------------|----|
| Исходный код программы Reverse Cipher.....    | 74 |
| Пробный запуск программы Reverse Cipher.....  | 74 |
| Ввод комментариев и установка переменных..... | 75 |
| Определение длины строки.....                 | 76 |
| Знакомство с циклом while.....                | 77 |
| Булев тип данных .....                        | 77 |
| Операторы сравнения .....                     | 78 |
| Блоки.....                                    | 81 |

|                                                                                                  |            |
|--------------------------------------------------------------------------------------------------|------------|
| Инструкция <code>while</code> .....                                                              | 82         |
| “Наращивание” строки .....                                                                       | 83         |
| Усовершенствование программы за счет использования<br>функции <code>input()</code> .....         | 86         |
| Резюме .....                                                                                     | 86         |
| <b>ГЛАВА 5</b>                                                                                   |            |
| <b>ШИФР ЦЕЗАРЯ</b>                                                                               | <b>89</b>  |
| Исходный код программы <code>Caesar Cipher</code> .....                                          | 90         |
| Пример выполнения программы <code>Caesar Cipher</code> .....                                     | 91         |
| Импорт модулей и установка переменных .....                                                      | 92         |
| Константы и переменные .....                                                                     | 93         |
| Цикл <code>for</code> .....                                                                      | 94         |
| Пример цикла <code>for</code> .....                                                              | 95         |
| Цикл <code>while</code> , эквивалентный циклу <code>for</code> .....                             | 95         |
| Инструкция <code>if</code> .....                                                                 | 96         |
| Пример инструкции <code>if</code> .....                                                          | 96         |
| Инструкция <code>else</code> .....                                                               | 97         |
| Инструкция <code>elif</code> .....                                                               | 98         |
| Операторы <code>in</code> и <code>not in</code> .....                                            | 99         |
| Строковый метод <code>find()</code> .....                                                        | 100        |
| Шифрование и дешифрование символов .....                                                         | 101        |
| Обработка “завертывания” символьного набора .....                                                | 102        |
| Обработка символов, не включенных в символьный набор .....                                       | 103        |
| Вывод и копирование преобразованной строки .....                                                 | 103        |
| Шифрование других символов .....                                                                 | 104        |
| Резюме .....                                                                                     | 105        |
| <b>ГЛАВА 6</b>                                                                                   |            |
| <b>ВЗЛОМ ШИФРА ЦЕЗАРЯ МЕТОДОМ ГРУБОЙ СИЛЫ</b>                                                    | <b>109</b> |
| Исходный код программы <code>Caesar Hacker</code> .....                                          | 110        |
| Пример выполнения программы <code>Caesar Hacker</code> .....                                     | 111        |
| Установка переменных .....                                                                       | 112        |
| Организация цикла с помощью функции <code>range()</code> .....                                   | 112        |
| Дешифрование сообщения .....                                                                     | 114        |
| Использование строкового форматирования для отображения<br>ключа и дешифрованных сообщений ..... | 115        |
| Резюме .....                                                                                     | 116        |

## ГЛАВА 7

### ШИФРОВАНИЕ С ПОМОЩЬЮ ПЕРЕСТАНОВОЧНОГО ШИФРА 119

|                                                                                              |     |
|----------------------------------------------------------------------------------------------|-----|
| Как работает перестановочный шифр.....                                                       | 120 |
| Шифрование сообщения вручную.....                                                            | 121 |
| Создание программы шифрования .....                                                          | 122 |
| Исходный код программы <code>Transposition Encrypt</code> .....                              | 123 |
| Пример выполнения программы <code>Transposition Encrypt</code> .....                         | 125 |
| Создание собственных функций с помощью инструкции <code>def</code> .....                     | 125 |
| Определение функции с параметрами .....                                                      | 126 |
| Изменение параметров сказывается лишь внутри функции .....                                   | 127 |
| Определение функции <code>main()</code> .....                                                | 128 |
| Передача ключа и сообщения в качестве аргументов.....                                        | 129 |
| Списковый тип данных .....                                                                   | 130 |
| Изменение элементов списка .....                                                             | 131 |
| Вложенные списки.....                                                                        | 132 |
| Применение функции <code>len()</code> и оператора <code>in</code> к спискам.....             | 133 |
| Конкатенация и репликация списков с помощью операторов <code>+</code> и <code>*</code> ..... | 134 |
| Алгоритм шифрования с помощью перестановочного шифра.....                                    | 134 |
| Составные операторы присваивания.....                                                        | 136 |
| Перемещение текущего индекса по строке сообщения .....                                       | 137 |
| Строковый метод <code>join()</code> .....                                                    | 139 |
| Возвращаемые значения и инструкция <code>return</code> .....                                 | 139 |
| Пример инструкции <code>return</code> .....                                                  | 140 |
| Возврат зашифрованного шифротекста .....                                                     | 140 |
| Переменная <code>__name__</code> .....                                                       | 141 |
| Резюме .....                                                                                 | 142 |

## ГЛАВА 8

### ДЕШИФРОВАНИЕ ПЕРЕСТАНОВОЧНОГО ШИФРА 145

|                                                                                           |     |
|-------------------------------------------------------------------------------------------|-----|
| Как дешифровать на бумаге текст, зашифрованный с помощью перестановочного шифра .....     | 146 |
| Исходный код программы <code>Transposition Decrypt</code> .....                           | 147 |
| Пример выполнения программы <code>Transposition Decrypt</code> .....                      | 149 |
| Импорт модулей и функция <code>main()</code> .....                                        | 149 |
| Дешифрование сообщения с помощью ключа.....                                               | 150 |
| Функции <code>round()</code> , <code>math.ceil()</code> и <code>math.floor()</code> ..... | 150 |
| Функция <code>decryptMessage()</code> .....                                               | 151 |
| Булевы операторы .....                                                                    | 153 |
| Настройка переменных <code>column</code> и <code>row</code> .....                         | 157 |

|                                         |     |
|-----------------------------------------|-----|
| Вызов функции <code>main()</code> ..... | 159 |
| Резюме .....                            | 159 |

## **ГЛАВА 9**

### **НАПИСАНИЕ ТЕСТОВ**

**161**

|                                                                                    |     |
|------------------------------------------------------------------------------------|-----|
| Исходный код программы <code>Transposition Test</code> .....                       | 162 |
| Пример выполнения программы <code>Transposition Test</code> .....                  | 163 |
| Импорт модулей.....                                                                | 164 |
| Создание псевдослучайных чисел.....                                                | 164 |
| Создание случайной строки.....                                                     | 166 |
| Дублирование строки произвольное число раз .....                                   | 167 |
| Списковые переменные используют ссылки .....                                       | 168 |
| Передача ссылок .....                                                              | 170 |
| Использование функции <code>copy.deepcopy()</code> для<br>дублирования списка..... | 171 |
| Функция <code>random.shuffle()</code> .....                                        | 171 |
| Случайное перемешивание строки.....                                                | 172 |
| Тестирование каждого сообщения.....                                                | 172 |
| Проверка корректности результата и завершение программы.....                       | 174 |
| Вызов функции <code>main()</code> .....                                            | 174 |
| Тестирование программы-тестера .....                                               | 175 |
| Резюме .....                                                                       | 175 |

## **ГЛАВА 10**

### **ШИФРОВАНИЕ И ДЕШИФРОВАНИЕ ФАЙЛОВ**

**177**

|                                                                                           |     |
|-------------------------------------------------------------------------------------------|-----|
| Простые текстовые файлы .....                                                             | 178 |
| Исходный код программы <code>Transposition File Cipher</code> .....                       | 178 |
| Пример выполнения программы <code>Transposition File Cipher</code> .....                  | 180 |
| Работа с файлами.....                                                                     | 181 |
| Открытие файлов .....                                                                     | 181 |
| Запись и закрытие файлов.....                                                             | 182 |
| Чтение из файла.....                                                                      | 183 |
| Функция <code>main()</code> .....                                                         | 183 |
| Проверка существования файла .....                                                        | 184 |
| Функция <code>os.path.exists()</code> .....                                               | 184 |
| Проверка существования файла с помощью функции<br><code>os.path.exists()</code> .....     | 185 |
| Строковые методы, используемые для повышения гибкости<br>пользовательского ввода .....    | 185 |
| Строковые методы <code>upper()</code> , <code>lower()</code> и <code>title()</code> ..... | 186 |
| Строковые методы <code>startswith()</code> и <code>endswith()</code> .....                | 186 |

|                                                                   |     |
|-------------------------------------------------------------------|-----|
| Использование строковых методов в программе .....                 | 187 |
| Чтение входного файла .....                                       | 188 |
| Измерение затрат времени на шифрование и дешифрование.....        | 188 |
| Модуль <code>time</code> и функция <code>time.time()</code> ..... | 188 |
| Использование функции <code>time.time()</code> в программе.....   | 189 |
| Запись в выходной файл.....                                       | 190 |
| Вызов функции <code>main()</code> .....                           | 190 |
| Резюме .....                                                      | 191 |

## ГЛАВА 11

### ПРОГРАММНОЕ РАСПОЗНАВАНИЕ АНГЛИЙСКИХ СЛОВ **193**

|                                                                                                     |     |
|-----------------------------------------------------------------------------------------------------|-----|
| Может ли компьютер понимать английский язык? .....                                                  | 194 |
| Исходный код модуля <code>Detect English</code> .....                                               | 196 |
| Применение модуля <code>Detect English</code> .....                                                 | 197 |
| Указания по использованию модуля и установка констант.....                                          | 198 |
| Словарный тип данных .....                                                                          | 198 |
| Различие между словарями и списками.....                                                            | 200 |
| Добавление и изменение элементов словаря .....                                                      | 200 |
| Применение функции <code>len()</code> к словарям .....                                              | 201 |
| Применение оператора <code>in</code> к словарям .....                                               | 202 |
| Поиск элементов в словарях выполняется быстрее,<br>чем в списках .....                              | 202 |
| Использование циклов <code>for</code> со словарями.....                                             | 203 |
| Реализация файла словаря .....                                                                      | 203 |
| Метод <code>split()</code> .....                                                                    | 204 |
| Разбивка файла словаря на отдельные слова .....                                                     | 204 |
| Возврат данных в виде словаря .....                                                                 | 205 |
| Подсчет количества английских слов в сообщении .....                                                | 206 |
| Ошибка деления на нуль.....                                                                         | 206 |
| Считаем английские слова.....                                                                       | 207 |
| Функции <code>float()</code> , <code>int()</code> и <code>str()</code> и целочисленное деление..... | 208 |
| Нахождение доли английских слов в сообщении .....                                                   | 209 |
| Удаление небуквенных символов .....                                                                 | 209 |
| Метод <code>append()</code> списка.....                                                             | 210 |
| Создание строки, объединяющей буквы.....                                                            | 211 |
| Распознавание английских слов .....                                                                 | 211 |
| Использование аргументов по умолчанию .....                                                         | 212 |
| Вычисление процентной доли .....                                                                    | 213 |
| Резюме .....                                                                                        | 215 |

## ГЛАВА 12

### ВЗЛОМ ПЕРЕСТАНОВОЧНОГО ШИФРА 217

|                                                                     |     |
|---------------------------------------------------------------------|-----|
| Исходный код программы <code>Transposition Hacker</code> .....      | 218 |
| Пример выполнения программы <code>Transposition Hacker</code> ..... | 219 |
| Импорт модулей.....                                                 | 220 |
| Создание многострочного текста с помощью тройных кавычек.....       | 221 |
| Отображение результатов взлома сообщения .....                      | 222 |
| Получение взломанного сообщения.....                                | 222 |
| Строковый метод <code>strip()</code> .....                          | 224 |
| Применение строкового метода <code>strip()</code> .....             | 225 |
| Невозможность взлома сообщения.....                                 | 226 |
| Вызов функции <code>main()</code> .....                             | 226 |
| Резюме.....                                                         | 226 |

## ГЛАВА 13

### АФФИННОЕ ШИФРОВАНИЕ С ПОМОЩЬЮ МОДУЛЬНОЙ АРИФМЕТИКИ 229

|                                                                           |     |
|---------------------------------------------------------------------------|-----|
| Модульная арифметика.....                                                 | 230 |
| Оператор деления с остатком .....                                         | 231 |
| Нахождение множителей для вычисления наибольшего<br>общего делителя ..... | 232 |
| Групповое присваивание .....                                              | 234 |
| Алгоритм Евклида для нахождения НОД.....                                  | 235 |
| Как работают мультипликативный и аффинный шифры.....                      | 236 |
| Выбор допустимых мультипликативных ключей.....                            | 237 |
| Шифрование с помощью аффинного шифра.....                                 | 238 |
| Дешифрование с помощью аффинного шифра.....                               | 239 |
| Вычисление модульных обращений.....                                       | 240 |
| Оператор целочисленного деления.....                                      | 241 |
| Исходный код модуля <code>Cryptomath</code> .....                         | 242 |
| Резюме.....                                                               | 243 |

## ГЛАВА 14

### ПРОГРАММИРОВАНИЕ АФФИННОГО ШИФРА 245

|                                                                        |     |
|------------------------------------------------------------------------|-----|
| Исходный код программы <code>Affine Cipher</code> .....                | 246 |
| Пример выполнения программы <code>Affine Cipher</code> .....           | 248 |
| Импорт модулей, настройка констант и функция <code>main()</code> ..... | 248 |
| Вычисление и проверка ключей.....                                      | 250 |
| Кортежи.....                                                           | 251 |
| Выявление слабых ключей .....                                          | 251 |
| Сколько ключей может иметь аффинный шифр.....                          | 253 |

|                                                                           |            |
|---------------------------------------------------------------------------|------------|
| Написание функции шифрования .....                                        | 255        |
| Написание функции дешифрования .....                                      | 256        |
| Генерирование случайных ключей.....                                       | 257        |
| Вызов функции <code>main()</code> .....                                   | 258        |
| Резюме .....                                                              | 259        |
| <b>ГЛАВА 15</b>                                                           |            |
| <b>ВЗЛОМ АФФИННОГО ШИФРА</b> .....                                        | <b>261</b> |
| Исходный код программы <code>Affine Hacker</code> .....                   | 262        |
| Пример выполнения программы <code>Affine Hacker</code> .....              | 263        |
| Импорт модулей, настройка констант и функция <code>main()</code> .....    | 264        |
| Функция взлома аффинного шифра .....                                      | 265        |
| Оператор возведения в степень .....                                       | 265        |
| Вычисление общего количества возможных ключей.....                        | 266        |
| Инструкция <code>continue</code> .....                                    | 267        |
| Использование инструкции <code>continue</code> для пропуска кода.....     | 268        |
| Вызов функции <code>main()</code> .....                                   | 269        |
| Резюме .....                                                              | 270        |
| <b>ГЛАВА 16</b>                                                           |            |
| <b>ПРОГРАММИРОВАНИЕ ПРОСТОГО ПОДСТАНОВОЧНОГО ШИФРА</b> .....              | <b>271</b> |
| Как работает простой подстановочный шифр .....                            | 272        |
| Исходный код программы <code>Simple Substitution Cipher</code> .....      | 273        |
| Пример выполнения программы <code>Simple Substitution Cipher</code> ..... | 275        |
| Импорт модулей, настройка констант и функция <code>main()</code> .....    | 276        |
| Списковый метод <code>sort()</code> .....                                 | 277        |
| Функции-обертки.....                                                      | 279        |
| Функция <code>translateMessage()</code> .....                             | 280        |
| Строковые методы <code>isupper()</code> и <code>islower()</code> .....    | 282        |
| Сохранение регистра букв с помощью метода <code>isupper()</code> .....    | 283        |
| Генерирование случайных ключей.....                                       | 284        |
| Вызов функции <code>main()</code> .....                                   | 285        |
| Резюме .....                                                              | 285        |
| <b>ГЛАВА 17</b>                                                           |            |
| <b>ВЗЛОМ ПРОСТОГО ПОДСТАНОВОЧНОГО ШИФРА</b> .....                         | <b>287</b> |
| Дешифрование с использованием словарных шаблонов.....                     | 288        |
| Поиск шаблонов слов.....                                                  | 288        |
| Поиск возможных вариантов дешифрования букв .....                         | 289        |
| Обзор процесса взлома .....                                               | 291        |
| Модуль <code>Word Patterns</code> .....                                   | 292        |



|                                                              |     |
|--------------------------------------------------------------|-----|
| Исходный код программы Simple Substitution Hacker .....      | 293 |
| Пример выполнения программы Simple Substitution Hacker ..... | 296 |
| Импорт модулей и настройка констант .....                    | 297 |
| Поиск символов с помощью регулярных выражений .....          | 298 |
| Функция main() .....                                         | 298 |
| Вывод результатов взлома .....                               | 299 |
| Создание словарей шифробукв .....                            | 300 |
| Создание пустого словаря шифробукв .....                     | 300 |
| Добавление букв в дешифровальный словарь .....               | 300 |
| Пересечение двух словарей .....                              | 302 |
| Как работают вспомогательные функции .....                   | 303 |
| Выявление достоверно установленных букв в словарях .....     | 307 |
| Тестирование функции removeSolvedLetterFromMapping() .....   | 309 |
| Функция hackSimpleSub() .....                                | 310 |
| Строковый метод replace() .....                              | 312 |
| Дешифрование сообщения .....                                 | 313 |
| Дешифрование сообщения в интерактивной оболочке .....        | 315 |
| Вызов функции main() .....                                   | 316 |
| Резюме .....                                                 | 316 |

## ГЛАВА 18

### ПРОГРАММИРОВАНИЕ ШИФРА ВИЖЕНЕРА

**319**

|                                                                    |     |
|--------------------------------------------------------------------|-----|
| Использование многобуквенных ключей в шифре Виженера .....         | 320 |
| Чем длиннее ключ шифра Виженера, тем он надежнее .....             | 322 |
| Выбор ключа, предотвращающего словарные атаки .....                | 323 |
| Исходный код программы Vigenere Cipher .....                       | 323 |
| Пример выполнения программы Vigenere Cipher .....                  | 325 |
| Импорт модулей, настройка констант и функция main() .....          | 325 |
| Создание строк с помощью списковых методов append() и join() ..... | 326 |
| Шифрование и дешифрование сообщения .....                          | 328 |
| Вызов функции main() .....                                         | 331 |
| Резюме .....                                                       | 331 |

## ГЛАВА 19

### ЧАСТОТНЫЙ АНАЛИЗ

**333**

|                                                                                            |     |
|--------------------------------------------------------------------------------------------|-----|
| Анализ частотности букв в тексте .....                                                     | 334 |
| Частотное соответствие букв .....                                                          | 336 |
| Вычисление оценки частотного соответствия букв<br>для простого подстановочного шифра ..... | 337 |
| Вычисление оценки частотного соответствия букв<br>для перестановочного шифра .....         | 338 |

|                                                                                   |     |
|-----------------------------------------------------------------------------------|-----|
| Использование частотного анализа в случае шифра Виженера .....                    | 339 |
| Исходный код программы <code>Frequency Analysis</code> .....                      | 340 |
| Сохранение букв алфавита в порядке <code>ETAOIN</code> .....                      | 341 |
| Подсчет букв в сообщении .....                                                    | 342 |
| Получение первого элемента кортежа.....                                           | 344 |
| Упорядочение букв, встречающихся в сообщении, в соответствии с частотностью ..... | 344 |
| Подсчет букв с помощью функции <code>getLetterCount()</code> .....                | 345 |
| Создание словаря счетчиков частотности со списками букв.....                      | 345 |
| Сортировка списков букв в порядке, обратном порядку <code>ETAOIN</code> .....     | 346 |
| Сортировка списков словаря по частотности .....                                   | 351 |
| Создание списка отсортированных букв .....                                        | 353 |
| Вычисление оценки частотного соответствия букв в сообщении.....                   | 353 |
| Резюме .....                                                                      | 355 |

## ГЛАВА 20

### ВЗЛОМ ШИФРА ВИЖЕНЕРА

**357**

|                                                                                       |     |
|---------------------------------------------------------------------------------------|-----|
| Использование перебора по словарю для взлома шифра Виженера методом грубой силы ..... | 358 |
| Исходный код программы <code>Vigenere Dictionary Hacker</code> .....                  | 358 |
| Пример выполнения программы <code>Vigenere Dictionary Hacker</code> .....             | 359 |
| О структуре программы .....                                                           | 360 |
| Использование метода Касиски для определения длины ключа .....                        | 361 |
| Нахождение повторяющихся сегментов .....                                              | 361 |
| Определение множителей в интервалах повторения .....                                  | 362 |
| Получение каждой <i>n</i> -й буквы строки.....                                        | 364 |
| Применение частотного анализа для взлома каждого подключа .....                       | 365 |
| Перебор возможных подключей методом грубой силы.....                                  | 367 |
| Исходный код программы <code>Vigenere Hacker</code> .....                             | 368 |
| Пример выполнения программы <code>Vigenere Hacker</code> .....                        | 374 |
| Импорт модулей и функция <code>main()</code> .....                                    | 375 |
| Нахождение повторяющихся последовательностей.....                                     | 375 |
| Вычисление множителей интервалов повторения .....                                     | 378 |
| Удаление дубликатов с помощью функции <code>set()</code> .....                        | 379 |
| Удаление дублирующихся множителей и сортировка списка .....                           | 380 |
| Нахождение наиболее часто встречающихся множителей .....                              | 381 |
| Нахождение наиболее вероятной длины ключа .....                                       | 383 |
| Списковый метод <code>extend()</code> .....                                           | 383 |
| Расширение словаря <code>repeatedSeqSpacings</code> .....                             | 384 |
| Извлечение множителей из списка <code>factorsByCount</code> .....                     | 385 |

|                                                                                  |     |
|----------------------------------------------------------------------------------|-----|
| Получение букв, зашифрованных одним и тем же подключом.....                      | 386 |
| Попытки дешифрования с помощью ключей вероятной длины .....                      | 387 |
| Аргумент <code>end</code> функции <code>print()</code> .....                     | 390 |
| Запуск программы в “тихом” режиме или с выводом информации для пользователя..... | 390 |
| Нахождение возможных комбинаций подключей .....                                  | 391 |
| Вывод дешифрованного текста с сохранением корректного регистра букв.....         | 395 |
| Получение взломанного сообщения.....                                             | 396 |
| Выход из цикла, если найден потенциальный ключ.....                              | 397 |
| Тестирование всех остальных вариантов длины ключа методом грубой силы.....       | 398 |
| Вызов функции <code>main()</code> .....                                          | 399 |
| Изменение значений констант, используемых в программе.....                       | 399 |
| Резюме .....                                                                     | 400 |

## **ГЛАВА 21**

### **ОДНОРАЗОВЫЙ ШИФРОБЛОКНОТ**

**403**

|                                                                  |     |
|------------------------------------------------------------------|-----|
| Не поддающийся взлому одноразовый шифроблокнот .....             | 404 |
| Выравнивание длины ключа по размеру сообщения .....              | 404 |
| Создание истинно случайного ключа .....                          | 406 |
| Избегайте двухразовых шифроблокнотов.....                        | 407 |
| Почему двухразовый шифроблокнот эквивалентен шифру Виженера..... | 408 |
| Резюме .....                                                     | 409 |

## **ГЛАВА 22**

### **НАХОЖДЕНИЕ И ГЕНЕРИРОВАНИЕ ПРОСТЫХ ЧИСЕЛ**

**411**

|                                                              |     |
|--------------------------------------------------------------|-----|
| Что такое простое число .....                                | 412 |
| Исходный код модуля <code>primeNum</code> .....              | 414 |
| Пример работы модуля <code>primeNum</code> .....             | 417 |
| Как работает алгоритм перебора делителей.....                | 417 |
| Реализация алгоритма перебора делителей.....                 | 419 |
| Решето Эратосфена .....                                      | 420 |
| Генерирование простых чисел с помощью решета Эратосфена..... | 422 |
| Тест Миллера – Рабина для проверки простоты числа .....      | 423 |
| Проверка больших простых чисел.....                          | 424 |
| Генерирование больших простых чисел.....                     | 426 |
| Резюме .....                                                 | 426 |

**ГЛАВА 23**  
**ГЕНЕРИРОВАНИЕ КЛЮЧЕЙ ДЛЯ КРИПТОСИСТЕМ**  
**С ОТКРЫТЫМ КЛЮЧОМ** **429**

|                                                              |     |
|--------------------------------------------------------------|-----|
| Криптосистемы с открытым ключом.....                         | 430 |
| Проблема аутентификации.....                                 | 432 |
| Цифровые подписи.....                                        | 432 |
| Остерегайтесь атак MITM.....                                 | 433 |
| Порядок генерирования открытых и закрытых ключей.....        | 434 |
| Исходный код программы Make Public Private Keys.....         | 435 |
| Пример выполнения программы Make Public Private Keys.....    | 437 |
| Создание функции main().....                                 | 438 |
| Генерирование ключей с помощью функции generateKey().....    | 439 |
| Вычисление значения $e$ .....                                | 439 |
| Вычисление значения $d$ .....                                | 440 |
| Возврат ключей.....                                          | 440 |
| Создание файлов ключей с помощью функции makeKeyFiles()..... | 441 |
| Вызов функции main().....                                    | 443 |
| Гибридные криптосистемы.....                                 | 443 |
| Резюме.....                                                  | 444 |

**ГЛАВА 24**  
**ПРОГРАММА ШИФРОВАНИЯ С ОТКРЫТЫМ КЛЮЧОМ** **445**

|                                                                                   |     |
|-----------------------------------------------------------------------------------|-----|
| Как работают криптосистемы с открытым ключом.....                                 | 446 |
| Создание блоков.....                                                              | 446 |
| Преобразование строки в блок.....                                                 | 447 |
| Арифметика шифрования и дешифрования с открытым ключом.....                       | 449 |
| Преобразование блока в строку.....                                                | 451 |
| Почему нельзя взломать сообщение, зашифрованное<br>с помощью открытого ключа..... | 452 |
| Исходный код программы Public Key Cipher.....                                     | 455 |
| Пример выполнения программы Public Key Cipher.....                                | 458 |
| Начало программы.....                                                             | 460 |
| Как программа определяет, что ей делать: шифровать<br>или дешифровать.....        | 460 |
| Преобразование строк в блоки с помощью функции<br>getBlocksFromText().....        | 462 |
| Функции min() и max().....                                                        | 463 |
| Сохранение блоков в переменной blockInt.....                                      | 463 |
| Дешифрование блоков с помощью функции getTextFromBlocks().....                    | 465 |
| Использование спискового метода insert().....                                     | 466 |

|                                                                    |            |
|--------------------------------------------------------------------|------------|
| Объединение элементов списка message в одну строку .....           | 467        |
| Функция encryptMessage () .....                                    | 467        |
| Функция decryptMessage () .....                                    | 468        |
| Чтение открытого и закрытого ключей из соответствующих файлов..... | 469        |
| Запись зашифрованного сообщения в файл .....                       | 469        |
| Дешифрование содержимого файла .....                               | 472        |
| Вызов функции main () .....                                        | 474        |
| Резюме .....                                                       | 474        |
| <b>ПРИЛОЖЕНИЕ А</b>                                                |            |
| <b>ОТЛАДКА КОДА PYTHON</b>                                         | <b>477</b> |
| Как работает отладчик .....                                        | 477        |
| Кнопка Go .....                                                    | 478        |
| Кнопка Step .....                                                  | 479        |
| Кнопка Over .....                                                  | 479        |
| Кнопка Out .....                                                   | 479        |
| Кнопка Quit .....                                                  | 479        |
| Отладка программы Reverse Cipher .....                             | 480        |
| Задание точек останова .....                                       | 482        |
| Резюме .....                                                       | 483        |
| <b>ПРИЛОЖЕНИЕ Б</b>                                                |            |
| <b>ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ</b>                               | <b>485</b> |
| <b>ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ</b>                                        | <b>503</b> |