

## Программирование сложных информационных систем

### Introduce

Ответ на вопрос по первому разделу предусматривает решение небольшой практической задачи на компьютере с использованием языка программирования по выбору абитуриента. При отсутствии установленных сред разработки разрешается использовать онлайн среду replit.com или аналогичную. Время на решение: 15 минут.

### Раздел 1. Информатика и программирование

#### 1. Системы счисления: кодирование чисел.

Реализация алгоритма перевода целого десятичного числа в двоичную систему счисления и обратно на одном из современных языков программирования.

#### 2. Асимптотика сложности алгоритмов.

Реализация алгоритма (по выбору) сортировки массива целых чисел на одном из современных языков программирования и анализ его сложности.

#### 3. Рекурсивные алгоритмы.

Реализация рекурсивной функции вычисления числа Фибоначчи с кэшированием.

#### 4. Структуры данных: массив, список, стек, очередь, множество, словарь, бинарное дерево.

Реализация алгоритма бинарного поиска в отсортированном массиве.

#### 5. Структуры данных: массив, список, стек, очередь, множество, словарь, бинарное дерево.

Реализация алгоритма проверки правильности правильной скобочной последовательности.

#### 6. Структуры данных: массив, список, стек, очередь, множество, словарь, бинарное дерево.

Реализация алгоритма поиска числа в бинарном дереве.

#### 7. Строки. Кодирование символов.

Реализация наивного алгоритма поиска подстроки в строке. Оценка асимптотики сложности алгоритма.

#### 8. Объектно-ориентированное программирование.

Инкапсуляция - пример реализации.

#### 9. Объектно-ориентированное программирование.

Наследование - пример реализации.

#### 10. Объектно-ориентированное программирование.

Полиморфизм - пример реализации.

#### 11. Хранение и обработка структурированных данных: csv-файлы.

Пример реализации чтения и обработки табличной информации из csv-файла.

#### 12. Обработка иерархически структурированных данных: json-файла.

Пример десериализации данных из json-файла и последующей сортировки по выбранному параметру.

#### 13. Рефакторинг кода. Признаки плохого кода.

Разделение кода на функции, модули, классы. Пример реализации поиска количества простых чисел в заданном диапазоне.