# Тест для соискателей вакансии программист

|  |
| --- |
| Уважаемый соискатель, перед Вами тест на вакансию программист в компании Saber Interactive. Индустрия видеоигр, в которой работает компания Saber, предъявляет высокие требования к качеству, эффективности и удобству восприятия программного кода. Поэтому мы предлагаем выполнить Вам 3 несложных задания, чтобы Вы могли продемонстрировать Вашу способность писать такой код.  Просьба не использовать в первых двух задачах библиотечные функции и классы, являющиеся решением этих задач (например stl::bitset). Для третьей задачи можно использовать стандартные функции и контейнеры из STL. |

*Фамилия, имя, отчество выполнившего тест:*

*Дата выполнения:*

*Примерное количество времени, затраченного на выполнение теста:*

## Задачи

1. Напишите функцию, которая принимает на вход знаковое целое число и печатает его двоичное представление, не используя библиотечных классов или функций.
2. Напишите функцию, удаляющую последовательно дублирующиеся символы в строке:

void RemoveDups(char\* str);

// пример использования

char data[] = “AAA BBB AAA”;

RemoveDups(data);

printf(“%s\n”, data); // “A B A”

1. Реализуйте функции сериализации и десериализации двусвязного списка в бинарном формате в файл. Алгоритмическая сложность решения должна быть меньше квадратичной.

// структуру ListNode модифицровать нельзя

struct ListNode {

ListNode \* prev;

ListNode \* next;

ListNode \* rand; // указатель на произвольный элемент данного списка, либо NULL

std::string data;

};

class List {

public:

void Serialize (FILE \* file); // сохранение в файл (файл открыт с помощью fopen(path, "wb"))

void Deserialize (FILE \* file); // загрузка из файла (файл открыт с помощью fopen(path, "rb"))

private:

ListNode \* head;

ListNode \* tail;

int count;

};

Примечание: сериализация подразумевает сохранение и восстановление полной структуры списка, включая взаимное соотношение его элементов между собой.

*Спасибо за уделенное время и выполнение нашего теста!*