

1. Простое задание (по вариантам).

Нарисуйте дерево и выпишите последовательность бит, получаемую при компрессии алгоритмом Хаффмана слова, заданного номером варианта. Нужно выписать только сжатую последовательность бит, без служебной информации.

Номер варианта	Слово для компрессии
1, 11, 21	МОТОРОТА
2, 12, 22	КОРРЕКТОР
3, 13, 23	ПЕРЕПЕЧКА
4, 14, 24	ПРОМОПОСТ
5, 15, 25	ПАРАЛЛЕЛЬ
6, 16, 26	АППАЛАЧИ
7, 17, 27	СКОРОГОВОРКА
8, 18, 28	МОСТООТРЯД
9, 19, 29	ЗАКАЗАТЬ
10, 20, 30	АПШАРЕЛЬ

2. Сложное задание (одно на всех, требует кодирования на C++).

Напишите программу **минимального размера** (по размеру исходного кода), которая выведет на экран указанный ниже код – часть кода разрабатываемого на кафедре симулятора для моделирования цифровых схем.

Требуется, чтобы полностью совпадал не просто текст, но и сохранялось форматирование текста (отступы, регистр и т.д.).

Вывод сделать в текстовый файл out.txt.

Мы будем просто сравнивать два текстовых файла и искать отличия.

В приведённом участке кода все отступы – это пробелы (уж не знаю, как PDF сохранит), в концах строк никаких лишних пробелов нет.

ВНИМАНИЕ!

1. От вас мы ждём **ТОЛЬКО ФАЙЛ .CPP** (специально написал заглавными буквами, капсом, и не говорите, что не видели). Кто пришлёт более одного файла – даже проверять не будем, будем считать, что решение не прислано.
2. Вы выполняете СРС – самостоятельную работу! В случае если будут присланы два одинаковых исходника, мы засчитываем первый присланный и не засчитываем второй. Поверьте мне на слово, мы сможем увидеть, что два исходника одинаковые (даже если вы возьмёте чужой исходник и, например, поменяете имена переменных). Или делайте самостоятельно, или не присылайте вовсе.

```
--- (начать вывод со следующей строки, это выводить не надо)
class gate_not : public gate {
public:
    gate_not(std::string name);
    void operate();
    bool postprocess();
};

class gate_and : public gate {
public:
    gate_and(std::string name);
    void operate();
    bool postprocess();
};

class gate_or : public gate {
public:
    gate_or(std::string name);
    void operate();
    bool postprocess();
};

class gate_nor : public gate {
public:
    gate_nor(std::string name);
    void operate();
    bool postprocess();
};

class gate_nand : public gate {
public:
    gate_nand(std::string name);
    void operate();
    bool postprocess();
};
--- (вывода заканчивается на предыдущей строке, это выводить не надо)
```

Участки текста со словами «это выводить не надо» выводить в файл не нужно, это ограничители, чтобы вам было понятнее, откуда и куда выводить.

В каждой группе мы выберем по одному самому короткому исходнику на автомат (при условии соблюдения всех обычных требований по выполнению лабораторных работ и СРС – на момент окончания семестра все СРС присылались, все ЛР сданы в срок).

Ответ нужно прислать в электронном виде на адрес dima@pkims.ru

В заголовке письма прошу указать группу, фамилию.

Последний день приёма ответов для этого задания – 7 ноября 23:59. Ответы, отправленные позже этого срока, например, 8 ноября в 0:01, рассматриваться не будут.