# Задача 1. (за полное решение – 30 баллов, 15 за алгоритм, 5 за временную оценку, 10 – за код)

Дано шаров цветов. Необходимо выбрать шаров таким образом, чтобы количество различных цветов среди выбранных шаров было максимально. Если существует несколько вариантов выборки – выбрать любой из них. Данные подаются на вход в виде двух строк, где первая строка содержит два числа – общее число шаров (первое) и число шаров, которые необходимо выбрать (второе). Вторая строка содержит чисел, обозначающих цвета шаров. На выходе программа распечатывает строку из чисел – цвета шаров, которые надо выбрать.

Необходимо разработать (в виде текстового описания, блок-схемы либо псевдокода) алгоритм для решения задачи. Дополнительные баллы – за временную оценку сложности и реализацию алгоритма на С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 5 3  1 1 1 2 2 | 1 1 2 |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 10 4  8 8 8 8 8 8 2 8 1 8 | 1 2 8 8 |

# Задача 2. (10 баллов)

Необходимо «нарисовать» рождественскую елку заданной высоты. На вход подается число, обозначающее высоту елки, на выходе- строки, «рисующие» елку. Программу реализовать на С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 1 | \* |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 5 | \*  \*\*  \*\*\*  \*\*\*\*  \*\*\*\*\* |

# Задача 3. (за полное решение – 20 баллов, 10 за алгоритм, 5 за временную оценку, 5 за код)

Два массива называются похожими, если совпадают множества чисел, находящихся в этих массивах. Требуется определить, похожи ли два массива. На вход подаются три строки:

1. Два числа и - длины массивов
2. чисел – первый массив
3. чисел – второй массив

На выходе необходимо распечатать слово «True», если массивы похожи и «False», если нет (в обоих случаях – без кавычек). Необходимо разработать (в виде текстового описания, блок-схемы либо псевдокода) алгоритм для решения задачи. Дополнительные баллы – за временную оценку сложности и реализацию алгоритма на С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 4 3  1 2 3 2  3 2 1 | True |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 2 3  1 2  2 1 3 | False |

# Задача 4. (20 баллов)

Дано АВЛ-дерево:

Является ли дерево сбалансированным? Объяснить почему. Если необходимо – сделать балансировку, рисуя результат на каждом шаге.

# Задача 5. (20 баллов, 10 за алгоритм, 5 за временнуб оценку, 5 – за код)

Вы живете в городе, где все улицы расположены в виде идеально квадратной сетки и проходят только с севера на юг и с запад на восток. Проход по улице от одного пересечения до другого всегда занимает ровно 1 минуту. Отправляясь на важную встречу вы внезапно пришли на условленное место (пересечение двух улиц) на минут раньше. Вы решили воспользоваться мобильным приложением, которое по запросу предоставляет замкнутый маршрут заданной продолжительности от точки, где вы находитесь. Маршрут приходит в виде строки следующего вида:

W E N S

где буквы обозначают соответствующие стороны света (W – запад, E – восток, N – север, S – юг), буквы разделены пробелами. Каждая буква задает направление, в котором нужно пройти один квартал.

Однако вы сомневаетесь в правильности работы приложения и хотите его проверить.

На вход подается две строки:

1. Целое число N – количество минут.
2. Массив из букв длиной N, содержащий предлагаемый маршрут.

На выход необходимо напечатать «True», если маршрут за N минут приведет вас обратно к той точке, где вы находитесь. В противном случае – напечатать «False».

Необходимо разработать (в виде текстового описания, блок-схемы либо псевдокода) алгоритм для решения задачи. Дополнительные баллы – за временную оценку сложности и реализацию алгоритма на С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 5  W W N N E | False |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 4  N W S E | True |

# Задача 6 (20 баллов)

Дан IP-адрес и маска подсети. Определить адрес сети и широковещательный адрес сети.

На вход подается строка из пяти чисел, где первые 4 числа – октеты IP-адреса в порядке от старшего к младшему, а 5-ое число – маска подсети. Например, строка

192 168 3 4 24

задает IP-адрес 192.168.3.4 с маской подсети 255.255.255.0.

На выходе должны быть три строки:

1. Адрес сети
2. Широковещательный адрес сети

Программа должна быть реализована на языке С, С++ или Java. Вводимые данные всегда корректны.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 192 168 9 2 24 | 192.168.9.0  192.168.9.255 |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 192 168 9 3 21 | 192.168.8.0  192.168.15.255 |

# Задача 7 (за полное решение – 30 баллов, 15 за алгоритм, 5 за временную оценку, 10 – за код)

Пусть центральной точкой массива называется элемент, для которого сумма всех элементов массива с меньшими индексами равна сумме всех элементов с большими индексами. Необходимо найти центральную точку в заданном массиве.

На вход подается 2 строки

1. Число , являющееся количеством элементов в массиве
2. целых чисел – элементы массива

На выходе необходимо напечатать индекс (индексы отсчитываются с 0) элемента, являющегося центральной точкой массива или -1 если такой элемент не существует. Необходимо разработать (в виде текстового описания, блок-схемы либо псевдокода) алгоритм для решения задачи. Дополнительные баллы – за временную оценку сложности и реализацию алгоритма на С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 3  -1 0 1 | -1 |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 5  4 1 7 9 3 9 | 3 |

# Задача 8(20 баллов)

Посчитать количество повторяющихся символов. На вход строка из символов, разделенных пробелами. Необходимо вывести количество символов в строке, которые встречаются более, чем 1 раз. Сравнение символов должно быть регистронезависимым. Реализовать программу необходимо на языке С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| A s d f | 0 |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| A s a g t t | 2 |

# Задача 9(10 баллов)

Панграмма – это последовательность символов алфавита, содержащая каждую букву алфавита как минимум один раз. Необходимо определить, является ли введенная строка панграммой или нет.

На вход подается строка из символов английского алфавита, пробелов и знаков препинания.

На выходе вывести «True», если строка является панграммой и «False» если нет. Реализовать программу необходимо на языке С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| Test test test | False |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| The quick brown fox jumps over the lazy dog | True |

# Задача 10(20 баллов)

Вы разрабатываете программное обеспечение для банкомата. Банкомат должен выдать запрошенную клиентом сумму минимальным количеством купюр по 1, 2, 5 и 10 кредитов.

На вход подается целое число - запрошенная клиентом сумма.

На выходе необходимо распечатать строку, содержащую 4 числа, где первое число – количество выдаваемых купюр по 1 кредиту, 2-ое – по 2, 3-е по 5, а четвертое – по 10 кредитов. Реализовать программу необходимо на языке С++, Java или Python.

Пример 1:

|  |  |
| --- | --- |
| Вход | Выход |
| 12 | 0 1 0 1 |

Пример 2:

|  |  |
| --- | --- |
| Вход | Выход |
| 3 | 1 1 0 0 |