

Задача А. Think outside the box

Вместо уроков информатики мальчик Вася занимался подготовкой к олимпиадам по программированию и достиг в этом больших успехов. К сожалению, в результате постоянных тренировок, Вася стал мыслить «внутри коробки».

Например, увидев задачу, приведенную ниже, Вася подумал: «Хм, странно, где ограничения? Ладно, напишу длинную арифметику, точно пройдет. Так, что же выбрать? Обратную польскую нотацию или рекурсивный разбор выражения?...». Конечно же, задача решается намного проще, чем думает Вася, но для этого надо мыслить «вне коробки» и ходить на школьные уроки информатики. А ограничения можно узнать, посмотрев на входной файл, который прилагается к условию.

Задача формулируется так:

Вам дан набор корректных арифметических выражений, в которых используются целые числа, математические операции $+$, $-$ и $*$ и круглые скобки. Запись каждого выражения начинается со знака $=$. Для каждого из выражений посчитайте результат вычислений.

В первом тесте записано 10 арифметических выражений. Оценка за этот тест: 30 баллов. За каждое правильно вычисленное выражение начисляется 3 балла. Проверка осуществляется в режиме on-line (результат виден сразу).

Во втором тесте записано 70000 арифметических выражений. Оценка за этот тест: 70 баллов. За каждое правильно вычисленное выражение начисляется 0.001 балла. Во время тура проверяется, что сданный файл содержит 70000 целых чисел, а также проверяется правильность вычисления первых 70 выражений. Проверка правильности ответа осуществляется в режиме off-line (результат виден после окончания тура).

Примеры

Входные данные	Результат
$=2+3$	5
$=2*3$	6
$=(2+3)*4$	20

Задача В. Плохой стиль программирования

У многих начинающих программистов дела со стилем написания кода обстоят очень плохо. Их девиз «Работает — и ладно!».

В настоящем программировании программы не только пишут, но и читают и даже исправляют. Хороший стиль программирования нужен для того, чтобы программу было легко читать и понимать.

В этой задаче вам требуется по арифметическому выражению без скобок, содержащему только однобуквенные переменные (обозначенные буквами от a до z) и знаки арифметических операций $+$, $-$, $*$ и $/$ определить, написано оно в хорошем стиле или нет. Выражения не содержат на своих концах пробелов или знаков арифметических операций, а также двух операций или названий переменных подряд.

Хорошим считается один из следующих стилей:

1. Все выражение записано без пробелов, например: $a/b+c*d$
2. Арифметические операции меньшего приоритета с двух сторон обособляются ровно одним пробелом, например: $a/b + c*d$
3. Все арифметические операции с двух сторон обособляются ровно одним пробелом, например: $a / b + c * d$

При этом в пределах одного выражения нужно придерживаться одного стиля. Например, выражение $a/b + c * d$ является смешением всех трех стилей и, как итог, его стиль плохой.

Для каждого из выражений выведите 1, если оно написано в одном из трех хороших стилей и 0 в противном случае.

В первом тесте записано 10 арифметических выражений. Оценка за этот тест: 30 баллов. За каждое правильно определенное выражение начисляется 3 балла. Проверка осуществляется в режиме on-line (результат виден сразу).

Во втором тесте записано 70000 арифметических выражений. Оценка за этот тест: 70 баллов. За каждое правильно определенное выражение начисляется 0.001 балла. Во время тура проверяется, что сданный файл содержит 70000 целых чисел, а также проверяется правильность определения первых 70 выражений. Проверка правильности ответа осуществляется в режиме off-line (результат виден после окончания тура).

Примеры

Входные данные	Результат
$a+b+c$	1
$a / b+c$	0
$a + b$	0
a/ b	0
$a * b / c$	1

Задача С. Кодовый замок

Кодовый замок открывается с помощью кода из N подряд идущих цифр в системе счисления с основанием K . Замок открывается как только была введена нужная последовательность цифр, при этом неважно, какие цифры были введены до этого. Допустимы последовательности с ведущими нулями.

Например, если код замка 11, то он может быть открыт последовательностями 11, 01101, 01011 и любыми другими последовательностями, содержащими 11.

Вася давно не был дома и забыл код от замка. Помогите ему составить как можно более короткую последовательность, которая содержит все возможные коды.

В первом тесте $N = 3$, $K = 2$. Максимальная оценка за этот тест: 30 баллов. Решения будут получать баллы по формуле $Score = \max(0, 30 \times \frac{24 - Solution}{24 - BestSolution})$, где $Solution$ — длина решения участника, а $BestSolution$ — длина кратчайшего решения, найденного участниками. Проверка осуществляется в режиме on-line (результат виден сразу).

Во втором тесте $N = 4$, $K = 10$. Максимальная оценка за этот тест: 70 баллов. Решения будут получать баллы по формуле $Score = \max(0, 70 \times \frac{40000 - Solution}{40000 - BestSolution})$, где $Solution$ — длина решения участника, а $BestSolution$ — длина кратчайшего решения, найденного участниками. Проверка правильности ответа осуществляется в режиме off-line (результат виден после окончания тура).

Примеры

Входные данные	Результат
2 2	00110

Выведенная строка содержит все двоичные числа длины 2: 00, 01, 10, 11.

Обратите внимание, что к этой задаче нет дополнительных входных файлов, все необходимые данные указаны в условии.

Задача D. Сжатие сообщения

По каналу связи передаются сообщения, содержащие только символы из алфавита размером N ; для передачи используется двоичный код, допускающий однозначное декодирование.

Для каждого символа из алфавита известно, сколько раз он встречается в сообщении. Сопоставьте каждому символу двоичный код так, чтобы длина закодированного сообщения получилась как можно меньше. Длины кодов различных символов могут не совпадать. Ни один код не должен являться началом кода другого символа.

В первой строке входного файла записано число N , в следующих N строках записан символ и сколько раз он встречается в сообщении (символ и число разделены табуляцией).

В выходной файл выведите все символы из алфавита и для каждого из них выведите сопоставленный ему двоичный код. Выводить символы можно в любом порядке. Если вариантов кодирования несколько, то вы можете вывести любой.

Качество решения (в формуле вычисления баллов это обозначено как *Solution*) — это длина закодированного сообщения в битах.

В первом тесте $N = 8$. Максимальная оценка за этот тест: 30 баллов. Решения будут получать баллы по формуле $Score = \max(0, 30 \times \frac{4350 - Solution}{4350 - BestSolution})$, где *Solution* — длина решения участника, а *BestSolution* — длина кратчайшего решения, найденного участниками. Проверка осуществляется в режиме on-line (результат виден сразу).

Во втором тесте $N = 64$. Максимальная оценка за этот тест: 70 баллов. Решения будут получать баллы по формуле $Score = \max(0, 70 \times \frac{6549282 - Solution}{6549282 - BestSolution})$, где *Solution* — длина решения участника, а *BestSolution* — длина кратчайшего решения, найденного участниками. Проверка правильности ответа осуществляется в режиме off-line (результат виден после окончания тура). Во время тура проверяется, что сообщение может быть однозначно декодировано.

Примеры

Входные данные	Результат
З	С 0
А 10	А 10
В 12	В 11
С 18	

Если код буквы А заменить на 1, то однозначное декодирование станет невозможным. Например, сообщение 111 может быть декодировано как АВ или ВА.