Разбор задач 2-го тура муниципального этапа

Всероссийской олимпиады школьников по информатике 2011 / 2012 учебного года

Задача A (F) Истинное наслаждение

Задача в стиле "на РОИ будет хуже".

Вот общая схема одного из решений.

Пусть в систему налито какое-то количество жидкости. Назовем группой сосудов максимальное (т.е. ни один нельзя добавить, не нарушив свойство) множество сосудов, попарно связанных трубками ниже уровня жидкости. Каждая такая группа характеризуется тем, что высота жидкости во всех сосудах в ней всегда будет оставаться одинаковой.

Сначала будем эмулировать процесс. Эмуляция будет происходить следующим образом: в каждый момент времени в стеке будем хранить информацию о всех группах сосудов, в порядке убывания высоты жидкости в них (можно показать, что никакие две группы сосудов никогда не имеют одинаковую высоту жидкости. Также можно показать, что наполняться будет самая низкая из таких групп). Заполним группу на вершине стека до максимальной высоты, пока не произойдет одно из двух:

1) Она выровняется по высоте с предыдущей в стеке. В этом случае надо будет объединить две группы на вершине в стеке после наливания.
2) Она достигнет уровня некоторой трубки до сосуда, который еще не начал заполняться. В этом случае этот сосуд образует новую группу (если быть точным, то когда в него зальется +eps жидкости). Просто добавим ее на вершину стека.

Во время эмуляции будет строить следующее бинарное дерево: вершиной его будут группы сосудов в какой-то момент времени, а группа, соответствующая вершине, получена путем объединения групп, соответствующих потомкам (это было в случае эмуляции 1). Случай эмуляции 2 будет соответствовать созданию новой вершины в дереве. В каждой вершине будем хранить количество элементов в соответствующей группе, с какой до какой высоты она наполнялось (это будет только один раз!) и в какой момент "времени" это произошло.

Теперь пусть к нам пришел некоторый запрос. При помощи, например, бинарного подъема мы можем найти группу, соответствующую сосуду и наполнению, а, значит, вычислить и ответ на него.

В решении есть несколько технических сложностей, которые могут быть преодолены:

1) В дереве нет вершины, которая соответствует высоте жидкости в сосуде 0. Поэтому, если в запросе высота 0, смело выдаем 0, не обращаясь к дереву.

2) Во время эмуляции неоходимо каким-то образом поддерживать быстрый ответ на минимальную высоту трубки до незаполненного сосуда из данной группы. Это можно делать, храня для каждой группы в очереди с приоритетами трубки до всех подходящих соседей (практически как в Дейкстре, только обновлять не надо), а при объединении групп перекидывать все элементы из меньшей очереди в большую (известно, что это суммарно O(M*log(N)) перекидываний в худшем случае).

Если все правильно сделать, то у нас будет O(N) событий эмуляции, которые в сумме обработаются за O(M * log(N) * log(M)), и потом Q запросов, каждый из которых будет обработан за O(log(N)). Итоговая сложность решения O((M*log(M)+Q)*log(N))

Задача B (G) Подручные средства

В задаче следовало обратить внимание на то, что единственным требованием является не попадать по шурупам. Это условие не запрещает, например, забить 7 гвоздей двумя ударами микроскопа, рассчитанного на забивание пяти гвоздей (три уже забитых гвоздя не пострадают).

Относительно же краев мы не можем получить какие-либо гарантии – достаточно ли там пространства, чтобы поместился микроскоп большего, а не необходимого размера. Внимательное рассмотрение первого примера из условия

```
14 2
3 5
OOXOOOOOOOXOOO
```

с ответом 5 приводит именно к такому выводу: гвозди с краев забивать надо либо микроскопом в точности подходящего размера, либо молотком.

Далее можно действовать “жадным” образом: для каждой “внутренней” последовательности гвоздей выбираем максимально возможный по размеру микроскоп и забиваем им гвозди, смещая его каждый раз таким образом, чтобы забить максимальное количество гвоздей и не затронуть при этом шурупы. Если не удается найти подходящий микроскоп, придется действовать молотком.

Задача C (H) Взаимное расположение

Задача носит скорее технический характер и предполагает аккуратное исполнение всех описанных команд с контролем пересечений и соприкосновений. Каждый очередной параллелепипед, помещенный на доску, должен иметь вокруг себя “слой” толщиной хотя бы в одну пустую клетку. Аккуратная реализация этого требования также позволит посчитать оставшиеся “совершенно свободными” клетки, если все команды размещают параллелепипеды согласно требованиям.

Задача D (I) Излишняя поспешность

Здесь следует рассмотреть два случая: попытаться поместить бачок горизонтально или вертикально (отверстием вверх, поскольку воздух считаем несжимаемым), посчитать, сколько топлива зальется в него в том и в другом случае, после чего выбрать наибольшее из значений. Можно заметить, что в обоих случаях бачок имеет смысл погружать до соприкосновения с дном. Конечно, при этом необходимо контролировать, можно ли поместить бачок горизонтально в принципе (он может оказаться слишком длинным).

В случае, если бачок погружается в топливо горизонтально, то объем залившегося в него топлива будет либо равен объему бачка (если он окажется погруженным в него полностью), либо объему цилиндра с основанием, равным сегменту круга. Когда же бачок погружается вертикально, необходимо учесть, что при погружении он вытесняет некоторое количество жидкости (закон Архимеда) и первоначальный уровень топлива в баке возрастает. Поэтому, даже если уровень топлива в баке был меньше высоты бачка, после погружения бачка некоторое количество топлива все же может в него залиться.

Задача E (J) Искренняя предусмотрительность

Для каждого разряда можно посчитать минимально возможную и максимально возможную сумму, которую могут дать цифры в нем. Понятно, что знак вопроса можно заменить любой цифрой от 0 до 9, если он не является старшим разрядом в своем числе, и любой цифрой от 1 до 9, если он таковым является. Сумма цифр в разряде определяет перенос в следующий разряд, который, в свою очередь, будет прибавляться к сумме, накапливаемой в следующем разряде. Определив эти значения (минимальную и максимальную сумму в каждом разряде и допустимые переносы), можно организовать обратный проход – от самого старшего разряда к самому младшему, выбирая подходящие (согласно результирующей сумме) значения. Если окажется, что не существует ни одного допустимого значения, рассмотрение можно прервать и вывести No solution.