Теоретические задачи

К задачам этой части нужно написать развернутое решение и сдать его (набранное в редакторе Word, в формате pdf или в виде текстового файла) в системе регистрации.

Задача T1. Рыцари и лжецы [D]

На острове живут рыцари и лжецы. Рыцари всегда говорят правду, лжецы всегда лгут. Однажды на острове случился такой случай: Антона спросили: “Вы рыцарь или лжец?”. Не расслышав ответа, кто-то в толпе спросил у Васи, своего соседа: “Что ответил Антон?”. Вася сказал: “Антон утверждает, что он рыцарь”. Слышавший этот разговор Сережа заявил: “Вася лжет”. Какие из утверждений про Антона, Васю и Сережу точно верны? Какие точно ложны? Какие могут оказаться и ложными и истинными?

1. Антон и Вася – рыцари, Сережа – лжец;
2. Антон – рыцарь, Вася и Сережа – лжецы;
3. вывода об Антоне сделать нельзя, Вася – рыцарь, Сережа – лжец;
4. вывода о Сереже сделать нельзя, Вася – рыцарь, Антон – лжец;
5. все трое лжецы.

Задача T2. Алгоритм Евклида [C’, C]

Вася считает с помощью алгоритма Евклида наибольший общий делитель двух чисел. Чтобы не запутаться, он на каждом шаге записывает новую пару чисел, которая у него получилась. Например, если он считает НОД(40, 6) у него получается примерно следующее:

40	6
6	4
4	2
2	0

(в левом столбце он всегда пишет большее число, а в правом — меньшее).

Нам попали в руки Васины записи, но часть чисел в них стерлась. Посмотрите, на то что осталось и для каждого вычисления постарайтесь понять, мог ли у Васи получиться такой результат или он где-то сделал ошибку.

а)

*	71
*	*
13	*
*	*
2	0

б)

*	19
*	*
11	*
2	0

в)

8	*
*	*
*	*
*	*
*	*
*	0

г)

*	28
*	*
21	*
*	0

Задача T3. Ребус [C’, C, B]

В каких системах счисления, основание которых не превосходит 10, ребус `ABC + B0A = CA0` имеет хотя бы одно решение?

Буквам `A`, `B`, `C` должны соответствовать различные цифры. Числа `ABC`, `B0A` и `CA0` должны быть корректными числами для данной системы счисления и не могут содержать ведущих нулей.

Для тех систем счисления, где ребус не имеет решения, приведите обоснование того, что это так.

Задача Т4. Жадность [В]

Рассмотрим такую задачу: дано N предметов, i-ый предмет имеет вес w_i и стоимость c_i. Необходимо выбрать некоторое подмножество предметов (возможно, пустое), так чтобы сумма весов не превосходила данного числа W, и при этом сумма стоимостей была максимально возможной для таких подмножеств. Например, для набора предметов с весами 10, 20, 30 и стоимостями 50, 40, 30 соответственно при $W = 55$ оптимальное подмножество состоит из предметов 1 и 2.

Рассмотрим три алгоритма решения такой задачи:

- Алгоритм А: Начинаем с пустого множества, на каждом шаге добавляем в множество самый дорогой (т.е. с наибольшим c_i) из предметов, вес которого не превосходит $W' = W - sW$, где sW - сумма весов уже добавленных в множество предметов.
- Алгоритм Б: Начинаем с пустого множества, на каждом шаге добавляем в множество самый легкий (т.е. с наименьшим w_i) из предметов, вес которого не превосходит $W' = W - sW$, где sW - сумма весов уже добавленных в множество предметов.
- Алгоритм В: Начинаем с пустого множества, на каждом шаге добавляем в множество предмет, вес которого не превосходит $W' = W - sW$, где sW - сумма весов уже добавленных в множество предметов, и величина c_i/w_i максимальна среди всех доступных предметов.

Всегда ли эти алгоритмы дают оптимальный ответ? Если да, то докажите это. Если нет, приведите пример, когда алгоритм работает неверно. Какие еще алгоритмы для решения этой задачи вам известны?

Задача Т5. Игры разума [В, А]

Три мудреца садятся в круг. На лоб каждому из них прикрепляется бумажка, на которой написана цифра 1 или 2 одним из трех цветов: красным, синим или зеленым. Каждый мудрец видит все бумажки, кроме своей. Ученик может сообщить **только одному** из мудрецов цвет его бумажки **либо** число на его бумажке. Ученик не имеет права сообщить неверную информацию. Мудрецы не имеют права общаться с учеником и друг с другом. Каждый из мудрецов не знает о том, какие выводы делают другие мудрецы (в частности, к какому ответу они приходят).

- а) могут ли ученик и мудрецы договориться между собой таким образом, чтобы после сообщения информации один из мудрецов (выбранный заранее) в любом случае смог узнать, какая бумажка (цвет и цифра) у него на лбу?
- б) могут ли ученик и мудрецы договориться между собой таким образом, чтобы после сообщения информации все мудрецы в любом случае смогли узнать, какая бумажка (цвет и цифра) у каждого из них на лбу?
- в) пункт а), но цифры на бумажке могут принимать значения 1, 2 или 3
- г) пункт б), но цифры на бумажке могут принимать значения 1, 2 или 3

Задача Т6. Запросы [A]

Набор целых неотрицательных чисел S изначально пуст. Необходимо обрабатывать запросы «добавить число x в набор», «удалить число x из набора», «вывести значение функции f для текущего набора». Запросы не известны заранее (перед получением информации о следующем запросе необходимо ответить на текущий).

Для каждого из вариантов функции f:

- а) SUM (сумма всех чисел)
- б) MIN (минимум среди всех чисел)
- в) GCD (наибольший общий делитель всех чисел)
- г) $MOST\_FREQUENT$ (число, встречающееся в наборе наибольшее количество раз, среди таких минимальное)
- д) KTH_MIN (k-ое в возрастающем порядке число в наборе, k — константа, одинаковая для всех запросов)
- е) VAR_KTH_MIN (k-ое в возрастающем порядке число в наборе, значение k указывается дополнительно для каждого запроса)

опишите способ решения задачи, позволяющий как можно более эффективно обрабатывать запросы; укажите и обоснуйте временную сложность обработки каждого запроса в зависимости от текущего количества чисел в наборе (сложность не должна зависеть от максимального значения чисел; можно предполагать, что все элементарные операции с числами выполняются за $O(1)$).

Задача Т7. Строки Фибоначчи [A]

Строки Фибоначчи определяются следующим образом: $F_0 = b$, $F_1 = a$, для $i > 1$ $F_i = F_{i-1}F_{i-2}$.

- а) Докажите, что для $i > 4$ строка $F_{i-2}F_{i-2}$ — префикс F_i.
- б) Докажите, что если при $i > 2$ откинуть от F_i два последних символа, то останется палиндром.

Задача Т8. Точки и прямые [A]

На плоскости заданы n точек и m прямых. Известно, что никакая данная прямая не проходит ни через одну из данных точек.

Необходимо для каждой из m прямых определить, можно ли среди данных точек выбрать две, находящиеся по разные стороны от прямой.

Опишите алгоритм, который решает данную задачу за время $O((n + m)\log(n + m))$, обоснуйте его корректность и оценку на время работы.