



ИЗУМРУД.СТУДЕНТ
ОЛИМПИАДА УРАЛЬСКОГО ФЕДЕРАЛЬНОГО УНИВЕРСИТЕТА



3101209088268

Титульный лист

Направление Естественные науки Инженерные науки
 Математика и информатика Социальные и
 Экономика и управление гуманитарные науки

Вариативный блок 1 2 3 4 5

Курс 1 2 3 4 5 отсутствует

Фамилия Н Е Ч А Е В

Имя Р О М А Н

Отчество А Н Д Р Е Е В И Ч

Дата рождения 0 4 0 6 2 0 0 2

Город участия Е К А Т Е Р Ц И Н Б У Р Г

Аудитория 2 0 1

Телефон + 7 9 0 2 6 3 2 9 9 7 1

Дата 0 5 0 2 2 0 2 4

Подпись

Пример
заполнения

А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф
Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я 1 2 3 4 5 6 7 8 9 0



ИЗУМРУД.СТУДЕНТ

ОЛИМПИАДА УРАЛЬСКОГО ФЕДЕРАЛЬНОГО УНИВЕРСИТЕТА



Проверочный лист

Заполняется участниками

Направление Естественные науки Инженерные науки
 Математика и информатика Социальные и гуманитарные науки
 Экономика и управление

Вариативный блок 1 2 3 4 5

Курс 1 2 3 4 5 отсутствует

Город участия **ЕКАТЕРИНБУРГ**

Заполняется организаторами

Количество доп. листов _____ Количество черновиков к проверке : _____

Время выхода с **13 15** до: **13 21**

Протокол проверки

Заполняется жюри

Номер задания	1	2	3	4	5	6	7	8	9	10
Балл члена жюри №1	8	20								
Балл члена жюри №2	8	20								

Итоговый балл **28**

Подпись члена жюри №1 **Филатова.**

Подпись члена жюри №2 *[Signature]*

Пример заполнения

А Б В Г Д Е Ж З И Й К Л М Н О П Р С Т У Ф
Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я 1 2 3 4 5 6 7 8 9 0



ИНВАРИАНТНАЯ ЧАСТЬ:

1. Заметим, что кол-во времени в течение которого лиса ест сыр образует бесконечную геометрическую прогрессию с $q = \frac{1}{2}$. По формуле суммы бесконечной геометрической прогрессии $S = \frac{1}{1-q} = 2$ мин. Соответственно лиса бесконечно много раз откусывает сыр, и каждый раз время её уменьшится на $\frac{1}{2}$.

2. Пусть S - кол-во сыра, которое лиса съедает за 2 минуты, тогда

$$S = \sum_{i=1}^{\infty} \frac{2}{n(n+2)}$$
 Перенесем знаменатель: $S = \sum_{i=1}^{\infty} \frac{2}{n(n+2)} = 2 \cdot \sum_{i=1}^{\infty} \frac{1}{n(n+2)} \stackrel{(*)}{=} 2 \cdot \sum_{i=1}^{\infty} \frac{1}{n(n+1)} = 2 \cdot \frac{1}{8} = \frac{1}{4}$

Итак, кол-во сыра, съеденного лисой за 2 минуты равно $\frac{1}{4}$, (P.S. если конечно сумма «эталонового» ряда равна $\frac{1}{8}$).

3. Пусть S_1 - кол-во сыра, которое досталось I медв.
 S_2 - кол-во сыра, которое досталось II медв.

Тогда $S_1 = M_1 - 1 - S = 4 - 1 - \frac{1}{4} = 3 - \frac{1}{4}$? Нет решения
 $S_2 = M_2 - S = 3 - \frac{1}{4}$?

Уточнение: внимательно 1 из S_1 подразумевает тем, что лиса откусывает сыр только после 1 минуты.

ИТОГ + 8 баллов



Бланк ответов

БЛОК 3:

Схема БД для учета посуды:

Таблица "тип посуды":

ID: PK, int
 название типа: varchar

Таблица "размер посуды":

ID: PK, int
 название размера: varchar
 длина: ~~int~~ double
 ширина: double
 высота: double

Таблица "статус":

ID: PK, int
 название-статуса: varchar

Таблица "посуда":

тип: FK, int
 размер: FK, int
 статус: FK, int
 ID: PK, int

Схема БД для загрузки посуды:

Таблица "мойка":
 дата начала: DATE
 дата окончания: DATE
 человек: FK, int
 ID, PK, int

Таблица "человек":
 ID: PK, int
 имя: varchar

5

SQL запрос о кол-во предметов посуды по видам:

~~SELECT COUNT(*) FROM "посуда" ORDER BY тип~~

SELECT COUNT(*) FROM "посуда" JOIN "тип посуды" ON "тип посуды".ID = ~~тип~~ ^{posuda.type}
 ORDER BY "тип посуды". название типа.

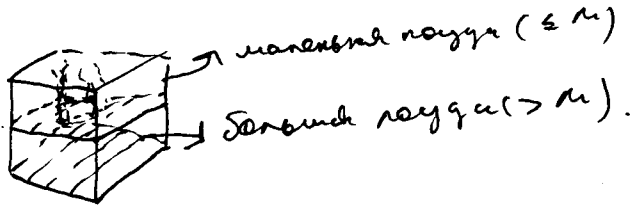
SQL запрос о кол-во моек:

SELECT COUNT(*) FROM "мойка"

5



Бланк ответов



Знаешь

База ~~знаний~~:

1. Если объем паузы $\leq m$, то помещаем паузу в верхн. палку, иначе нижнюю (некоторые размеры S, m, L - определяются по величине объема. Такого табличку не сложно сделать).
2. Все столбовые приборы помещаем в специальную ячейку ^{на} в верхней палке.
3. Если имеется несколько ~~таких~~ пауз? одинакового типа, ~~то~~ то упорядочиваем их в соседние ячейки в порядке дачной паузы.
4. Определим, что в передней части машины пауза находится антекавивней, чем в задней. Определим коэф. степени затравки от 0 до 1, где 0 - чистая, 1 - абсолютно грязная пауза. Тогда, если коэф ≤ 0.4 то убавляем паузу в заднюю часть, > 0.4 - в переднюю.

Для уменьшения задержек используем структуру данных «Priority Queue» - очередь с приоритетами. Каждой паузе назначим вес от 0 до 1 в зависимости от степени. Тогда упорядочиваем структуру будет кратчайшую паузу по правилу FIFO, учитывая степень затравки.
 ↓ (first in first out).

10

Алгоритм ~~на основе~~ ^{облегчен} ~~сложности~~.

Алгоритм:

1. Для каждой фрагмента паузы определить приоритет и поместить в Priority Queue.
2. Достанем элемент из Priority Queue пока очередь не пуста.

