

Задание по направлению «Информатика и вычислительная техника»

Профиль: «Компьютерные системы и сети»

КОД – 050

Время выполнения задания – 180 мин.

Задача 1 – 20 баллов.

У одного султана было 2 визиря. Захотел он проверить насколько они сообразительны. Позвал их обоих и сказал: Я загадал 2 числа от 2 до 100. Вы должны их мне назвать. При этом он сказал первому визирю произведение этих чисел, а второму - их сумму.

Первый визирь подумал и говорит:

- Я не знаю, что это за числа

На что второй ответил:

- Я был в этом уверен.

Тогда первый говорит:

- В таком случае я знаю, что это за числа.

Второй:

- Тогда и я знаю, что это за числа.

Назовите 2 числа, которые загадал султан.

После ответа первого становится ясно, что загаданные числа не являются оба простыми, а также если одно из них простое, то меньше 50. После ответа второго, из вариантов возможных сумм можно вычеркнуть все четные числа, как представимые в виде суммы двух простых чисел (спасибо Гольдбаху), а также числа вида простое+2. Можно вычеркнуть и нечетные числа, большие 53, т.к. их можно представить в виде большое простое число + остаток. (остались также числа больше 195, но они тоже, по понятным причинам, не подходят)

Итак, возможных вариантов сумм осталось 11: 11, 17, 23, 27, 29, 35, 37, 41, 47, 51, 53

Далее можно поступить следующим образом:

Записываем эти варианты в один столбец. Возле каждого, в строчку, выписываем варианты произведений для всех возможных разбиений на сумму 2-х слагаемых. Варианты произведений, повторяющиеся в нескольких строках отмечаем. (это те произведения, по которым первый НЕ смог бы определить числа при своем втором ответе).

Строка, в которой находится ЕДИНСТВЕННОЕ неотмеченное число, соответствует искомой сумме чисел, а неотмеченное число - искомое произведение.

Для суммы это получается 17, произведение = 52, искомые числа 13 и 4.

Задача 2 – 20 баллов.

Дана прямоугольная плоская поверхность, разделенная на равные прямоугольники. Прямоугольники разделены веревкой, которую муравьям неприятно переползать. В один из прямоугольников высыпается куча муравьев. Муравьям очень не нравится толчея и они хотят равномерно распределиться по поверхности. Муравьи в данном прямоугольнике имеют возможность оценить количество муравьев в четырех соседних прямоугольниках, после чего отправить в них сколько-то муравьев. Совещаться между собой или передавать информацию муравьи с разных прямоугольников не могут. Муравьи являются дружным коллективом, поэтому они не выгоняют муравьев, пришедших с соседних прямоугольников. По этой же причине они отправляют муравьев на соседние клетки только по команде Главного Муравья.

Помогите муравьям составить алгоритм, действуя по которому они смогут

максимально равномерно распределиться по поверхности. Решение обязано сходиться за конечное число шагов. Написание программы является плюсом.

Замечание 1. Главный Муравей работает как тактовый генератор, давая сигналы муравьям на всех клетках. Это решение значительно проще, чем если считать, что при отсутствии информации Главный Муравей дает команды разным клеткам.

Замечание 2. Решение сойдется быстрее, если учитывать, что и другие клетки пришлют своих муравьев в нашу, а не только мы отправим их.

Замечание 3. Основной задачей было избежать заикливания. Например, если число муравьев равно $k \cdot n + 1$, где n – целое число, а k – количество клеток, то при неверном решении одинокий муравей будет вечно странствовать по клеткам, так как он всегда будет «лишним». Необходимо учесть, среди прочего, эту и подобные ей ситуации.

Пример решения.

Считаем число муравьев, которые придут к нам из более населенных клеток как сумму разниц, деленную пополам. Округление при этом делаем в меньшую сторону, чтобы учесть вечно странствующего муравья. Прибавляем полученную разницу к количеству муравьев в нашей клетке, прибавляем число муравьев в клетках, где муравьев меньше, чем у нас. Делим полученную сумму на $m+1$, где m – число клеток, где муравьев меньше, чем у нас. Это число муравьев, которое должно получиться в этих m клетках после перераспределения. Вычисляем число муравьев, которых надо отправить в другие клетки.

Процесс пойдет быстрее, если учитывать разницу между прогнозируемым числом муравьев в соседней клетке и реально получившимся. Однако это может привести к заикливанию или возникновению положительной обратной связи при неаккуратном решении.

Задача 3 – 20 баллов.

Укажите назначение демилитаризованной зоны (DMZ) в сети. Приведите вариант структуры сети с DMZ. Обоснуйте решение и опишите функции структурных компонентов такой сети (в свободной форме).

См., например, [https://ru.wikipedia.org/wiki/DMZ_\(компьютерные_сети\)](https://ru.wikipedia.org/wiki/DMZ_(компьютерные_сети))

Задача 4 – 20 баллов.

Дан файл, содержащий список всех форм слов русского языка. В одной строчке записана одна форма слова. Формы слов сгруппированы по начальным формам слов, разделителем групп является пустая строка. Если у слова одна и та же форма может встретиться несколько раз («мамы» является родительным падежом единственного числа или именительным падежом множественного числа), она будет записана только один раз.

Требуется найти самое короткое (по числу операторов) и самое быстрое решение следующей задачи. Язык для реализации может быть выбран любой (учтите, что естественные языки не включают в себя операторы).

Необходимо найти все формы слов, встречающиеся только у одного слова. Например, форма «стекло» не должно попасть в выходной список, так как оно является формой слов «стекать» и «стекло». Форма «защитниками» должна войти в список, так как оно является формой единственного слова «защитник».

Решение. Отсортировать все формы слов, удалить повторяющиеся.

Самое короткое решение можно получить с использованием bash:

sort file.txt | uniq -u

Самое быстрое - на компилируемом языке реализовать, например, быструю сортировку и удаление повторяющихся элементов.

Задача 5 – 20 баллов.

Написать алгоритм, моделирующий N вложенных арифметических циклов без использования рекурсии. В качестве входных данных задано количество циклов N . Для каждого из N циклов должны быть заданы свои начальные и конечные значения переменной цикла и шаг.

Создаем массив счетчиков размером N и инициализируем начальными значениями. Выполняем действие и увеличиваем, например, самый правый элемент массива на единицу. Если этот элемент достиг максимального значения, сбрасываем его в начальное значение и увеличиваем на единицу следующий элемент. Если и он достиг максимального значения повторяет сброс и переход к следующему элементу. Работа прекращается, когда самый левый элемент достиг максимального значения.