|  |
| --- |
| 1. **Спектакли**.   В некотором зале в один день запланированы *n* спектаклей. Для каждого спектакля известен интервал времени (*begin*, *end*). Требуется распланировать максимальное количество спектаклей таким образом, чтоб они не пересекались. |
| 1. **Автогонка**   На автогонке на пути от старта до финиша расположены *n* бензозаправок на разных дистанциях друг от друга. При полном баке машина конкурсанта может проехать *d* км. Конкурсант стремится проехать весь путь от старта к финишу, останавливаясь на дозаправку бензином минимальное количество раз. Составьте оптимальный план остановок конкурсанту. |
| 1. **Группы студентов**   В текстовый файл записаны фамилии и имена студентов и их средние баллы из *х* разных групп. Каждая группа содержит минимум *a* и максимум *b* студентов(2<a<b<30). Студентов необходимо перегруппировать по *x* студенов следующим образом: из каждой группы студентов берем по одному студенту, чтобы средний бал новой сформированной группы был равен 8. Вывести на эран списки каждой группы, перечислив фамилии и имена студентов. |
| 1. **Дискотека**   На дискотеке мальчиков *M* и *N* девочек. Определите все возможные случаи, когда мальчики могут пригласить девочек на танец таким образом, чтобы два мальчика не могли пригласить одну и ту же девочку на танец, и чтобы каждый мальчик танцевал.(*N*>=*M*). |
| 1. **Семинар**   На семинаре должны говорить 5 человек (*A, B, C, D, E*). Выведите все возможные списки выхода на трибуну, при условии, что человек *В* всегда говорит после *А*. |
| 1. **Охотники**   В некотором населенном пункте 15 охотников. Они делятся на группы по 3 человека таким образом, чтобы в течении 7 дней охоты никакие охотники не попадались в одну и ту же группу более одного раза. |
| 1. **Выборы**   Президент страны избирается *n* депутатами парламента. Президент может быть избран только в случае набора 2/3 голосов. Между некоторыми депутатами существуют конфликты интересов. Два депутата, между которыми существует конфликт интересов, голосуют по-разному. Если известно число *n* и пары чисел (*x,y*) представляющие собой порядковые номера депутатов, между которыми существует конфликт интересов, определите возможно или нет выбрать президента. |
| 1. **Триколор**   Имеются 6 разных цветов. Определите сколько возможных триколоров можем из них составить при условии, что посередине может быть либо желтый либо зеленый цвет и все три цвета разные. |
| 1. **Фальшивая монета**   Среди 1000 монет одна самая легкая. Определите фальшивую монету минимальным числом взвешиваний. |
| 1. **Конференция**   На конференции, чтобы отвести больше времени на неформальное общение, различные секции разнесли по разным аудиториям. Учёный с чрезвычайно широкими интересами хочет посетить несколько докладов, проходящих в разных секциях. Известно начало *si* и конец *fi* каждого доклада. Определить, какое максимальное количество докладов можно посетить. |
| 1. **Занятия**   Даны *n* заявок на проведение занятий в некоторой аудитории. В каждой заявке указаны начало и конец занятия (*si* и *fi* для *i*-й заявки). В случае пересечения заявок можно удовлетворить лишь одну из них. Заявки с номерами *i* и *j* совместны, если интервалы [*si,* *fi*) и [*sj,* *fj*) не пересекаются (то есть *fi ≤ sj* или *fj* ≤ *si*). Задача о выборе заявок состоит в том, чтобы набрать максимальное количество совместных друг с другом заявок. |
| 1. **Лестница**   По пути в школу Михаела должна подниматься по лестнице из ***n*** ступенек. Одним своим шагом она может преодолевать, по желанию, одну или две ступеньки. Будучи любознательной личностью, ученица поставила перед собой задачу: сколько существуют способов подъема по лестнице.  **Требование**. Написать программу, которая поможет Михаеле определить число **Nm** возможных способов подъема по лестнице, т.е. число последовательностей допустимых длин шага (по одной или по две ступеньки).  **Ввод.** С клавиатуры вводитсянатуральное ***n* –**число ступенек лестницы.  **Вывод.** На экран выводится **Nm –**число способов подъема по лестнице. |
| 1. **Домино**   Вывести на экран самую длинную цепочку костяшек домино, которую можно построить из *n* имеющихся. Эти костяшки сделаны из двух частей, на каждой из которых написано число от 1 до 6. Две костяшки можно соединить в цепочку только в том случае, если первое число второй костяшки совпадает со вторым числом первой костяшки.  Например, для *n=*6 костяшек, на которых написаны числа 4 2, 1 2, 4 4, 1 3, 2 5, 6 5 получим цепочку 3 1 2 4 |
| 1. **Ряд**   В классе *n* учеников, из них *f* девочек(1≤*f≤n*). Девочки пронумерованы от 1 до *f,*  а мальчики от *f+*1 до *n.* Определите все возможные способы формирования ряда из *p* учеников (*p ≤ n*) таким образом, чтобы никакие две девочки и никакие два мальчика не стояли рядом. |
| 1. **Точки**   На плоскости даны *n* точек *P*1, *P*2, ..., *Pn* c целыми координатами. Определите способы раскраски точек минимальным количеством цветов таким образом, чтобы две точки, находящиеся на расстоянии больше *D*, были раскрашены в разный цвет. |
| 1. **Совершенное число**   Число называется совершенным, если оно равно сумме его делителей отличных от самого числа. Например, 6=1+2+3, 28=1+2+4+7+14. Найти все совершенные числа меньше заданного числа **N** |
| 1. **Конь**   На шахматной доске в первой строке в первом столбце находится конь. Составьте план перемещения коня по шахматной доске таким образом, чтобы его ходы покрыли всю шахматную доску и ни в какой клетке он не побывал бы дважды. |
| 1. **Шаги вправо или вниз**   Дано прямоугольное поле размером клеток. Можно совершать шаги длиной в одну клетку вправо или вниз. Посчитать, сколькими способами можно попасть из левой верхней клетки в правую нижнюю. |
| 1. **Шаги вправо и по диагонали**   Дано прямоугольное поле размером клеток. Можно совершать шаги длиной в одну клетку вправо, вниз или по диагонали вправо-вниз. В каждой клетке записано некоторое натуральное число. Необходимо попасть из верхней левой клетки в правую нижнюю. Вес маршрута вычисляется как сумма чисел со всех посещенных клеток. Необходимо найти маршрут с минимальным весом. |
| 1. **Подпоследовательность**   Дана последовательность целых чисел. Необходимо найти ее самую длинную строго возрастающую подпоследовательность. |
| 1. **Распределение средств**   Для двух предприятий выделено *a* единиц средств. Как распределить все средства в течение 4 лет, чтобы доход был наибольшим, если известно, что доход от *x* единиц средств, вложенных в первое предприятие, равен *f*1(*x*), а доход от *y* единиц средств, вложенных во второе предприятие, равен *f*2(*y*). Остаток средств к концу года составляет *g*1(*x*)для первого предприятия и *g*2(*y*) для второго предприятия. Задачу решить методом динамического программирования.  https://www.matburo.ru/Examples/mp_dp/img1-7.gif |
| 1. **Квадрат**   Дана матрица элементы которой 0 и 1. Найти квадратную подматрицу максимальной площади на главной диагонали которой только 1. |
| 1. **Прямоугольник**   Дана матрица элементы которой целые числа. Найти прямоугольную подматрицу, сумма элементов которой по углам максимальна. |
| 1. **Телефонная сеть**   Имеются *n* городов, *n* ≤ 100, которые с помощью кабеля необходимо соединить в телефонную сеть. Для каждого города *i* известны его декартовы координаты *xi*, *yi*. Кабель, соединяющий два города, не может иметь ответвлений. Абоненты телефонной сети связываются между собой напрямую или через телефонные станции промежуточных городов. Длина кабеля, соединяющего города *i*, *j*, равна расстоянию между ними. Найдите суммарную минимальную длину кабеля, необходимого для того, чтобы между любыми двумя абонентами существовала телефонная связь. |
| 1. **Башни**.   Даны *n* прямоугольных плит, пронумерованных от 1 до *n*. Каждая плита *i* описывается толщиной *hi* и длинами сторон *xi*, *yi*. Напишите программу, которая находит максимальную высоту башни, которую можно сложить из рассматриваемых плит. Для обеспечения устойчивости башни необходимо соблюдать следующие правила (рис.):  – плиты можно размещать друг на друге только горизонтально, не на ребра или иным другим способом;  – соответствующие ребра укладываемых плит должны быть параллельны друг другу;  – любая плита башни должна целиком размещаться на поверхности нижней по отношению к ней плиты (естественно, плита в основании башни размещается на земле);  – часть плит может остаться неиспользованной.    Рисунок Башня, сложенная из прямоугольных плит  *Ввод*. Текстовый файл TURNURI.IN содержит в первой строке число *n*. В каждой из следующих *n* строк содержится по три целых положительных числа *xi, yi, hi*, разделенных пробелами.  *Вывод*. Текстовый файл TURNURI.OUT должен содержать в одной строке единственное целое число – максимальную высоту башни.  *Пример*.    Ограничения: *n* ≤ 1000; *xi, yi, hi* < 1000. |
| 1. **Сад**.   План некоторого сада прямоугольной формы размерами *n*≤*m* состоит из квадратных участков со сторонами, равными 1. На каждом участке растут деревья только одной породы. Любая порода может занимать один и более участков, причем не обязательно соседних. Напишите программу, которая находит прямоугольную область минимальной площади, содержащую не менее *k* пород деревьев. Стороны области должны совпадать со сторонами участков на плане. |
| 1. Фонари.   Парк прямоугольной формы разделен на квадраты одинакового размера. В любом из квадратов парка может быть установлен фонарь. Каждый фонарь освещает не только квадрат, в котором он находится, а также восемь соседних квадратов. Напишите программу, которая вычисляет минимальное количество фонарей, необходимых для освещения всего парка. |
| 1. **Пересечение прямоугольников.**   Даны *n* прямоугольников (*n* ≤ 10), стороны которых параллельны осям координат, а координаты вершин являются натуральными числами из множества {0, 1, 2, ..., 20}. Напишите программу, которая вычисляет площадь фигуры, получаемой в результате пересечения рассматриваемых прямоугольников. |
| 1. **Объединение прямоугольников**.   Даны *n* прямоугольников (n ≤ 10), стороны которых параллельны осям координат, а координаты вершин являются действительными числами. Напишите программу, которая вычисляет площадь фигуры, получаемой в результате объединения *n* прямоугольников. |
| 1. **Компас**.   Рассматриваются *n* точек на декартовой плоскости. Каждая точка *i* задана ее координатами *xi*, *yi*. Напишите программу, которая проверяет, можно ли нарисовать окружность с центром в одной из имеющихся точек так, чтобы она прошла через все остальные точки. |
| 1. **Психология**.   Имеется n служащих, *n* ≤ 100, которых необходимо распределить по m бригадам. Каждая бригада состоит из *k* служащих, *k* · *m* = *n*. Совместимость между служащими *i*, *j* характеризуется коэффициентом *rij* , который может принимать следующие целые значения: 0 (полная несовместимость), 1, 2, ... , 10 (отличная совместимость). Совместимость всего коллектива *C* вычисляется суммированием коэффициентов *rij* для всех возможных пар (*i*, *j*) внутри каждой бригады. Найдите максимальную совместимость *Cmax*, которую можно обеспечить подходящим распределением служащих по бригадам. |
| 1. **Лазер**.   Дана прямоугольная пластина размерами , где *m* и *n* натуральные числа. Пластина должна быть разрезана на маленьких пластин размером каждая. Поскольку исходная пластина неоднородна, то для каждого ее участка указывается плотность *dxy*, где *x*, *y* – это координаты нижнего левого угла соответствующего квадрата.  Для операции разрезания используется лазер. Каждая операция разрезания включает в себя:  – установку разрезаемой пластины на крепежном столе;  – установку мощности лазерного луча в соответствии с плотностью разрезаемого материала;  – одноразовое перемещение лазера вдоль прямой, параллельной одной из осей координат (собственно разрезание);  – снятие двух полученных пластин с крепежного стола.  Стоимость одной операции разрезания определяется по формуле c = *dmax*, где *dmax* – это максимальная из плотностей кусков , по границам которых прошел луч лазера. Очевидно, что общая стоимость *T* может быть получена путем сложения стоимостей с всех операций разрезания, необходимых для получения пластин размером . Напишите программу, которая находит минимальную стоимость *T*. |
| 1. **Кафе**.   Посетители кафе пронумерованы от 1 до *n*. Сначала только одному из них, посетителю с номером *i*, известна очень важная новость, которую он позже сообщает своим друзьям. В дальнейшем, любой посетитель *j*, который уже узнал эту новость, также сообщает ее только своим друзьям. Напишите программу, которая определяет число участников *p*, которые узнали указанную новость. Отношение “друзья” на множестве посетителей задается с помощью *m* различных пар типа {*j*, *k*} со значением “посетители *j*, *k* являются друзьями”. Считается, что 3 ≤ n ≤1000 и 2 ≤ m ≤ 30000. |