ПРОФИЛЬ «СИСТЕМЫ СВЯЗИ И ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ»

Профиль «Системы связи и дистанционного зондирования Земли» посвящен решению задач космической инженерии: расчету траекторий посадки аппаратов на поверхность планет Солнечной системы, конструированию и программированию полета спутников на околоземной орбите. Профиль включает в себя задачи по трем школьным предметам: математика, физика и информатика.

§1 Первый отборочный этап

Первый отборочный тур проводится индивидуально в сети Интернет, работы оцениваются автоматически средствами системы онлайн-тестирования. Для каждого из параллелей (9 класс или 10-11 класс) предлагается свой набор задач по физике, математике и информатике. Решение задач по информатике предполагало написание программ, допускалось использовать различные языки программирования: C/C++, Python, Java, C#. На решение задач учащимся отводилось 2 суток. Первый этап состоял из двух независимых попыток с разными задачами, участник мог решать только одну попытку (в случае, если он решал обе, в зачет шла лучшая). Решение каждой задачи дает определенное количество баллов. Баллы зачисляются в полном объеме за правильное решение задачи. Участники получают оценку за решение задач в совокупности по всем предметам данного профиля.

1.1 Первая попытка
Задачи по математике (9 класс)

Задача 1.1.1 (2 балла)

Условие:

Найдите последнюю цифру числа 2017^{2016}

Решение:

Число 2017 оканчивается цифрой 7; число 2017^2 – цифрой 9 ($7\times 7=49$); число 2017^3 – цифрой 3 ($7\times 9=63$); число 2017^4 – цифрой 1; число 2017^5 – цифрой 7, и далее последние цифры будут повторяться с периодичностью 4. Так как $2016=504\times 4$, то последней цифрой числа 2017^{2016} будет 1.

Ответ:

1

Задача 1.1.2 (2 балла)

Условие:

Вася отправился из пункта *M* со скоростью $V_B = 62$ км/ч через некоторое время после выезда Пети, скорость которого $V_П = 55$ км/ч. Они должны были прибыть в пункт *N* одновременно, но, пройдя 2/3 пути, Петя снизил скорость вдвое, и поэтому Вася догнал его уже на расстоянии $S = 124$ км от *N*. Найти расстояние от *M* до *N* (в км).

Решение:

Обозначим расстояние от *M* до *N* через *x* км, тогда разница во времени выезда равна $(\frac{x}{V_П} - \frac{x}{V_В})$ ч. Вася догнал Петю на расстоянии S км от N, т.е. проехал (x-S) км, затратив $\frac{x-S}{V_В}$ ч, а Петя затратил $(\frac{2x}{3V_П} + \frac{\frac{1}{3}x-S}{\frac{1}{2}V_П})$. Получаем уравнение для разницы во времени: $(\frac{x}{V_П} - \frac{x}{V_В}) = (\frac{2x}{3V_П} + \frac{\frac{1}{3}x-S}{\frac{1}{2}V_П}) - \frac{x-S}{V_В}$, откуда $x = 3S(2 - \frac{V_П}{V_В})$ км

Ответ:
$x = 3S(2 - \frac{V_П}{V_В})$ при произвольных V_B, $V_П$, S

Задача 1.1.3 (а) - 1 балл, б) - 1 балл, в) -2 балла)
Условие:
а) Жители планеты A утверждают, что $12 + 13 = 30$. Если считать, что они не ошибаются, а просто пользуются системой счисления с другим основанием, то чему в их системе равно $30 - 3$?
б) Чему равно 2×4 в системе счисления на планете B, если для них верно $34 + 44 = 111$?
в) Сколько существует натуральных чисел, четырехзначных в системе планеты A, трехзначных в системе планеты B и являющихся точным квадратом?

Решение:
а) Пусть жители планеты A пользуются системой счисления с основанием a $(a\leq3)$.
Тогда $a+2+a+3=3a$, откуда $a=5$.
Тогда $(30)_5 - (3)_5=(15)_{10} - (3)_{10}=(12)_{10}=(2\times5+2)_{10}=(22)_5$
б) Пусть жители планеты B пользуются системой счисления с основанием b $(b\leq4)$.
Тогда $(3b+4)+(4b+4)=b^2+b+1$, откуда находим (с учетом того, что $b\geq0$) $b=7$.
Тогда $(2\times4)=(8)_{10}=(7+1)_{10}=(11)_7$
в) Пусть число n удовлетворяет условию, тогда n четырехзначное в пятеричной системе, то есть $125\leq n < 625$. Так как n – трехзначное число в семеричной системе, то $n < 343$. Этому условию удовлетворяют 7 точных квадратов:
144,169, 196, 225, 256, 289 и 324.

Ответ:
а) 22
б) 11
в) 7

Задача 1.1.4 (2 балла)
Условие:
На сторонах AB и BC треугольника ABC взяли точки K и L соответственно так, что AK=KB/2, BL=LC. Отрезки CK и AL пересекаются в точке O. Найдите площадь треугольника ABC, если площадь треугольника AKO равна S.

Решение:
Пусть точка M – середина отрезка BK. Тогда ML – средняя линия в треугольнике KBC, значит ML ∥ KC. Тогда по теореме Фалеса AO=OL. Тогда имеем $S_{ABO}=3S_{AKO}$; $S_{ABL}=2S_{ABO}=6S_{AKO}$; $S_{ABC}=2S_{ABL}=12S_{AKO}=12\times S$

Ответ:
$12\times S$ при произвольной S_{AKO}

1.2 Первая попытка
Задачи по математике (10-11 класс)

Указание: *Во всех задачах, где это необходимо, число* π *считать приближенно равным 3,14*

Задача 1.2.1 (2 балла)

Условие:

Найдите наименьшее значение параметра a, при котором система имеет единственное решение:

$$\begin{cases} x + y + z = x^2 + 2y^2 \\ 3x + 4y + 5z = a \end{cases}$$

Решение:

Вычтем первое уравнение, умноженное на 5, из второго. Тогда система эквивалентна уравнению $5x^2+10y^2-2x-y=a$.

Выделим полные квадраты: $5(x-\frac{1}{5})^2+10(y-\frac{1}{20})^2=a+\frac{9}{40}$

Отсюда видно, что уравнение (а значит и исходная система) имеет единственное решение только при $a = -\frac{9}{40}$

Ответ:

- 0,225

Задача 1.2.2 (2 балла)

Условие:

На плоскости нарисовали выпуклый пятиугольник $A_1A_2A_3A_4A_5$. Затем построили точку A'_1, симметричную точке A_1 относительно A_2, точку A'_2, симметричную точке A_2 относительно A_3,… точку A'_5, симметричную точке A_5 относительно A_1. Известны координаты $(x_i';y_i')$ точек A'_i: $(x_1';y_1')$; $(x_2';y_2')$; $(x_3';y_3')$; $(x_4';y_4')$; $(x_5';y_5')$. Восстановите по ним абсциссу точки A_1.

Решение:

Пусть O – произвольная точка внутри пятиугольника $A_1A_2A_3A_4A_5$, обозначим $\overline{OA_k} = \overline{a_k}$, $\overline{OA_k'} = \overline{a_k'}$ $(k=1,...,5)$.

Тогда из условия следует, что $\overline{a_1} + \overline{a_1'} = 2\overline{a_2}$; $\overline{a_2} + \overline{a_2'} = 2\overline{a_3'}$; $\overline{a_3} + \overline{a_3'} = 2\overline{a_4'}$; $\overline{a_4} + \overline{a_4'} = 2\overline{a_5}$; $\overline{a_5} + \overline{a_5'} = 2\overline{a_1}$. Запишем систему уравнений только для абсцисс:

$$\begin{cases} x_1 + 2 = 2x_2 \\ x_2 + 12 = 2x_3 \\ x_3 + 13 = 2x_4 \\ x_4 + 6 = 2x_5 \\ x_5 - 4 = 2x_1 \end{cases}$$

Это система из 5 линейных уравнений с 5 неизвестными, решив ее, получаем:

$$x_1 = \frac{1}{31}(x_1' + 2x_2' + 4x_3' + 8x_4' + 16x_5')$$

Ответ:

$x_1 = \frac{1}{31}(x_1' + 2x_2' + 4x_3' + 8x_4' + 16x_5')$ при произвольных x_i'

Задача 1.2.3 (а) - 1 балл, б) - 1 балл, в) - 2 балла)

Условие:

а) Жители планеты A утверждают, что $22 + 33 = 110$. Если считать, что они не ошибаются, а просто пользуются системой счисления с другим основанием, то чему в их системе равно 22×33 ?

б) Чему равно $13+14$ в системе счисления на планете *B*, если для них верно $13\times14=215$?

в) Найдите натуральное число, являющееся точным квадратом, которое в системе планеты *A* четырехзначное, а в системе планеты *B* записывается одинаковыми цифрами. (Ответ дать в десятичной системе).

Решение:

а) Пусть жители планеты *A* пользуются системой счисления с основанием a $(a\leq3)$. Тогда $2a+2+3a+3=a^2+a$, то есть $5a+5=a^2+a$, откуда находим (с учетом того, что $a\geq0$) $a=5$.

Тогда $(2\times5+2)\times(3\times5+3)=(12)_{10}\times(18)_{10}=(216)_{10}=(1\times125+3\times25+3\times5+1)_{10}=(1331)_5$

б) Пусть жители планеты *B* пользуются системой счисления с основанием b $(b\leq5)$. Тогда $(b+3)\times(b+4)=2b^2+b+5$, откуда находим (с учетом того, что $b\geq0$) $b=7$.

Тогда $(1\times7+3)+(1\times7+4)=(10)_{10}+(11)_{10}=(21)_{10}=(3\times7)_{10}=(30)_7$

в) Так как искомое число (назовем его n) четырехзначное в пятеричной системе, то

$125\leq n<625$. Так как $7^2<125$, а $2\times7^3=686>625$, то возможны 2 варианта:

- n – трехзначное число в семеричной системе. Тогда $n=49x+7x+x=57x$. Но т.к. n является точным квадратом, то оно должно делиться на 57^2, что невозможно.
- n – четырехзначное число 1111_7. Проверим: $7^3+7^2+7=1=400=20^2$

Ответ:

а) 1331 б) 30 в) 400

Задача 1.2.4 (2 балла)

Условие:

Ученые наблюдали в течение 6 секунд за инородным телом, которое двигалось прямолинейно с переменной скоростью $v(t)=13+\sqrt{6t-t^2}$. Найдите путь, пройденный телом за это время.

Указание: Можно использовать тот факт, что путь, пройденный телом, численно равен площади фигуры под графиком зависимости скорости тела от времени.

Решение:

Преобразуем выражение $v(t)=13+\sqrt{6t-t^2}=13+\sqrt{9-(t-3)^2}$. То есть график функции $v(t)$ при $t\in[0;6]$ представляет из себя полуокружность радиуса 3 с центром в точке (3;13). Следовательно, площадь под графиком является суммой площадей полукруга радиуса 3 и прямоугольника со сторонами 13 и 6, т.е.

$S=13\times6+\frac{9\pi}{2}=78+14{,}13=92{,}13$

Ответ:

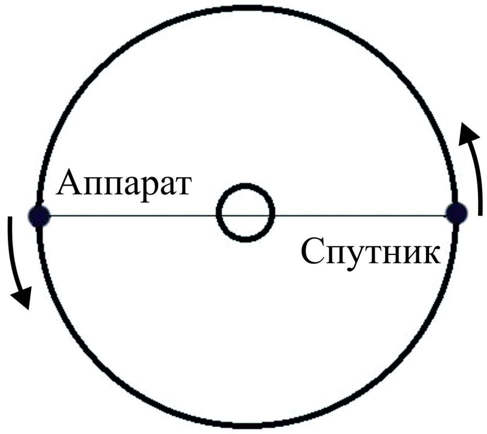
92, 13

1.3 Первая попытка
Задачи по физике (9 класс)

Задача 1.3.1 (2 балла)

Условие:

За какое минимальное время космический аппарат догонит геостационарный спутник, не сходя с орбиты спутника и двигаясь в направлении его движения. До начала движения спутник и аппарат находятся в диаметрально противоположных точках орбиты. Аппарат оснащен двигателями с силой тяги 1017Н, масса аппарата 90кг. Ответ выразите в минутах.



Решение:

Чтобы продолжать двигаться по той же орбите с большей скоростью, аппарат должен своими двигателями создавать дополнительный вклад в центростремительное ускорение.

Тогда второй закон Ньютона может быть записан как:

$\frac{mV^2}{R} = \frac{mgR_y^2}{R^2} + F$, где R - расстояние между центром Земли и высотой геостационарной орбиты, V - скорость аппарата, m - его масса, F - сила тяги двигателей. Тогда наибольшая скорость, которую может развить аппарат:

$V = \sqrt{\frac{gR_y^2}{R} + \frac{FR}{m}}$, скорость спутника в свою очередь можно оценить зная (или вычислив) высоту геостационарной орбиты $U = \frac{2\pi R}{T}$, где $T = 24$ч.

Искомое время τ может быть найдено из уравнения для пройденного спутником расстояния:

$$V\tau = U\tau + \pi R$$

и равно:

$$\tau = \frac{\pi R}{\sqrt{\frac{gR_y^2}{R} + \frac{FR}{m}} - \frac{2\pi R}{T}} = 120 \text{ мин.}$$

Ответ:

120 мин. +-0,01

Задача 1.3.2 (2 балла)

Условие:

Телеметрия спутника показала, зависимость координат спутника от времени, представленную в таблице.

Время (Московское)	Широта (˚)	Долгота (˚)	Высота над уровнем моря (км)
01:00	0	37,6 в.д.	8601
03:00	0	108,8 в.д.	8605
04:00	0	179,9 в.д.	8597

05:00	0	108,9 з.д	8598
06:00	0	37,7 з.д	8602
07:00	0	33,5 в.д.	8607

(в.д. - восточной долготы, з.д. - западной долготы).

Спутник должен визуально обнаружить аппарат, движущийся по геостационарной орбите. Он может это сделать, если спутник и аппарат на одной прямой друг с другом и центром Земли, и при этом аппарат движется относительно спутника со скоростью не более 2 км/с. Может ли его определить спутник? Для ответа на этот вопрос запишите скорость аппарата относительно спутника в км/с.

Геостационарный спутник - спутник вращающийся в плоскости экватора Земли с угловой скоростью равной угловой скорости вращения Земли. Высота геостационарной орбиты 35786 км.

Решение:

По предложенной таблице можно понять, проанализировав значения широты что спутник движется так же, как и геостационарный аппарат по экваториальной орбите. Из анализа долготы понятно, что спутник движется с постоянной угловой скоростью, т.е. по круговой орбите.

В данном случае, задачу можно решать двумя способами. Можно взяв за среднюю высоту спутника над уровнем моря 8600 км напрямую рассчитать его угловую скорость, а можно получить угловую скорость разделив изменения долготы на время, за которое они произошли.

Оба варианта дадут угловую скорость спутника $\omega \approx 3{,}45 \cdot 10^{-4}$ рад/с. При этом спутник движется по направлению вращения Земли, т.е. в туже сторону, что и геостационарный аппарат.

Линейная скорость спутника может быть вычислена, как $V = \omega \cdot (H + R_з)$, где H - высота полета спутника, а $R_з$ - радиус Земли. Эта скорость равна приблизительно 5.2км/с относительно центра Земли.

Линейную скорость геостационарного аппарата легко найти, если разделить длину окружности, по которой он движется, на 24 часа, важно только заметить, что к высоте геостационарной орбиты нужно еще добавить радиус Земли. Скорость аппарата будет равна примерно 3.1 км/с

Таким образом ,относительная скорость аппарат и спутника $\Delta V \approx 2.1$ км/с и спутник сможет визуально обнаружить аппарат.

Ответ:

$\Delta V \approx 2.1 \pm 0.02$ км/с.

Задача 1.3.3 (2 балла)

Условие:

Энергосистема спутника состоит из постоянной полезной нагрузки, солнечной батареи и аккумулятора. Информация о зависимости от времени потребляемой и вырабатываемой мощности за один оборот спутника представлена в таблице. Оцените по этой информации, какое количество энергии выработала солнечная батарея за один оборот спутника. Период обращения этого спутника 96 минут. Ответ дайте в кДж.

Время	Мощность прироста (+) и мощность расхода (-) энергии аккумулятора	Время	Мощность прироста (+) и мощность расхода (-) энергии аккумулятора
9:00	-5Вт	10:01	-5Вт
9:05	-5Вт	10:05	-5Вт
9:10	-5Вт	10:10	-5Вт
9:15	+5Вт	10:15	-5Вт
9:20	+5Вт	10:20	-5Вт
9:35	+5Вт	10:25	-5Вт
9:40	+5Вт	10:30	-5Вт
9:45	+5Вт	10:35	-5Вт
9:50	+5Вт	10:40	-5В
9:55	+5Вт	10:45	-5Вт

Решение:

Видно, что таблица захватывает чуть больше одного периода обращения спутника. Примерно половину времени с 9:15 до 10:00 происходит перевыработка 5Вт мощности, а вторую половину с 10:00 до 10:45 - недостаточная выработка электроэнергии в -5Вт. Можно предположить, что выработка связана с нахождением на солнечной стороне, а недостаток с нахождением с темной стороны.

Можно сделать вывод, что постоянная нагрузка $P = -N_-$ = 5Вт, тогда солнечная батарея вырабатывает $N = N_+ + P$ = 10Вт.

Энергию солнечная батарея вырабатывает половину периода обращения, т.е. в течение 48 минут. *E=10*48*60*Дж = 28,8кДж.

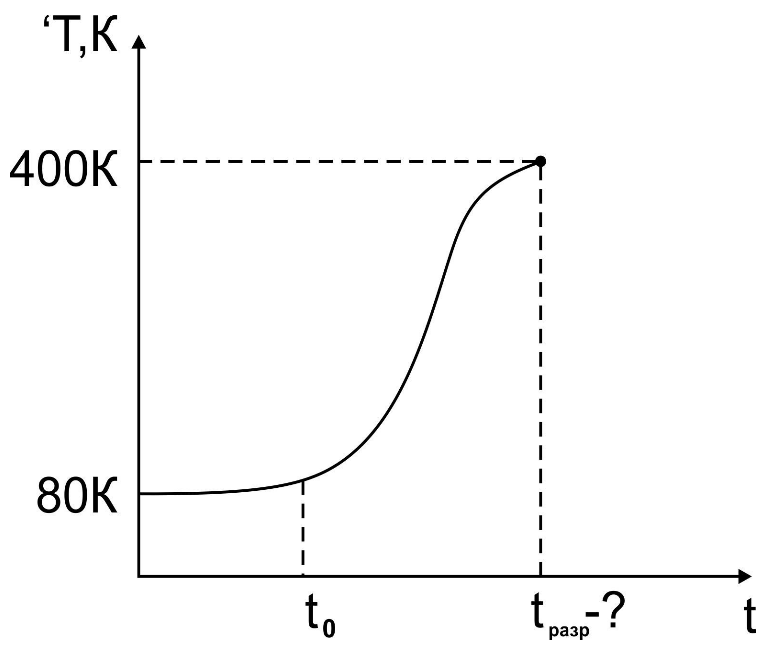
Ответ:

28,8 кДж.

Задача 1.3.4 (2 балла)

Условие:

Меркурий в 3 раз ближе к Солнцу, чем Земля. Оцените, за какое время разрушился бы космический аппарат на солнечной стороне Меркурия от перегрева, если теплоемкость теплопоглотителя аппарата равна 500кДж/°С, а коэффициент отражения поверхности аппарата α = 15%. Примерный график иллюстрирующий изменение температуры спутника приведен на рисунке.



Запишите время прошедшее с момента начала нагрева в минутах.

Солнечная постоянная – суммарная мощность солнечного излучения, *проходящего через единичную площадку, ориентированную перпендикулярно потоку солнечного излучения на расстоянии равном расстоянию от Солнца до Земли равна 1367Вт/м^2.*

Решение:

На единичную площадку на Меркурии попадает в 9 раз больше солнечной энергии, чем на Земле. Этот результат можно получить, если учесть, что полная мощность выделяемая Солнцем идет уходит в полную сферу, а площадь такой сферы падает пропорционально квадрату ее радиуса.

Тогда легко записать уравнение теплового баланса:

$(1-\alpha)P(t_{кр}-t_0)=C(T-T_0)$. t_0 - судя по графику время начала нагревания. $t_{кр}$ - время разрушения аппарата. Тогда искомое время T_0 =80К - начальная температура, а T =400К - конечная.

$(t_{кр}-t_0)=\dfrac{C(T-T_0)}{9K(1-\alpha)}$, где K- Солнечная постоянная. $(t_{кр}-t_0)$ = 1,53*10^{4}с = 255 минут.

Ответ:

255 минут.

Задача 1.3.5 (2 балла)

Условие:

Серебристый объект движется по орбите отличающей от орбиты МКС на 55 м и при этом вращается вокруг оси, перпендикулярной плоскости орбиты. Солнечный зайчик от объекта пробегает вдоль МКС за 1 секунду. С какой угловой скоростью вращается объект? Ответ выразите в радианах в секунду. *Длина МКС примерно 110м.*

Решение:

Отражение от объекта движется по МКС со скоростью $V=\dfrac{L}{\tau}$, где L - длина МКС, которую проходит отражение, а τ - время, за которое это происходит.

Угловая скорость движения зайчика вокруг спутника равна $\omega=\dfrac{V}{R}$, где R - расстояние между объектом и МКС.

При повороте зеркала на определенный угол отражение поворачивается на угол вдвое больший, соответственно угловая скорость вращения отражение в два раза выше угловой скорости вращения спутника. $\omega_{сп}=\dfrac{\omega}{2}=\dfrac{L}{2\tau R}$ = 1 рад/с

Ответ:

1 рад/с

1.4 Первая попытка
Задачи по физике (10-11 класс)

Задача 1.4.1 (2 балла)

Условие:

Телеметрия спутника показала, зависимость координат спутника от времени, представленную в таблице.

Время (Московское)	Широта (˚)	Долгота (˚)
01:00	0	37,6 в.д.
03:00	0	87,9 в.д.
05:00	0	138,3 в.д.
07:00	0	171.4 з.д
09:00	0	121,1 з.д
11:00	0	70,8 з.д.

(в.д. - восточной долготы, з.д. - западной долготы).

Спутник должен визуально обнаружить аппарат, движущийся по геостационарной орбите. Видеокамера спутника может поворачиваться за объектом с угловой скоростью не больше, чем $5\cdot10^{-5}$ рад/с. Чтобы определить сможет ли видеокамера заснять аппарат, найдите с какой относительной угловой скоростью он будет двигаться относительно спутника в тот момент, когда спутник, аппарат и центр Земли окажутся на одной прямой? Ответ выразите в радианах в секунду.

Решение:

По предложенной таблице можно понять, проанализировав значения широты что спутник движется так же, как и геостационарный аппарат по экваториальной орбите. Из анализа долготы понятно, что спутник движется с постоянной угловой скоростью, т.е. по круговой орбите.

Угловую скорость движения спутника можно найти из изменения долготы: $\omega = \dfrac{\Delta\varphi}{\Delta t}$, в данном случае она равна $\omega \approx 1{,}26\cdot10^{-4}$ рад/с или 0,0072 град/с.

Зная угловую скорость можно определить высоту орбиты спутника (относительно центра Земли).

$\omega^2 R = \dfrac{GM}{R^2} = \dfrac{gR_3^2}{R^2} \Rightarrow R = \sqrt[3]{\dfrac{gR_3^2}{\omega^2}} \approx 30\cdot10^3$ км. Тогда легко можно определить линейную скорость аппарата $V = \omega R \approx 3{,}7$ км/с.

Линейную скорость геостационарного аппарата легко найти, если разделить длину окружности, по которой он движется, на 24 часа, важно только заметить, что к высоте геостационарной орбиты нужно еще добавить радиус Земли. Скорость аппарата будет равна примерно 3.1 км/с

Угловую скорость вращения аппарата относительно спутника можно найти, если считать, что аппарат движется вокруг покоящегося спутника. Тогда его линейная скорость $\Delta V = V - V_{апп} = 3{,}7 км/с - 3{,}1 км/с = 0{,}6 км/с$, расстояние между ними в этот момент $\Delta H = (35{,}8 - 30{,}0)\cdot 10^3 км = 0{,}8\cdot 10^3 км$, тогда относительная угловая скорость: $\omega_{отн} = \frac{\Delta V}{\Delta H} \approx 4{,}7\cdot 10^{-5}$ рад/с.

Т.е. скорости слежения камеры спутника хватит, чтобы рассмотреть аппарат.

Ответ:

$\omega_{отн} \approx 4{,}7\cdot 10^{-5}$ рад/с.

Задача 1.4.2 (2 балла)

Условие:

Космонавт, гуляющий по небольшому сферическому астероиду из чистейшего водяного льда, решил выпрыгнуть на его орбиту и летать на высоте 20 км над астероидом. С какой минимальной скоростью он должен прыгнуть? Радиус астероида R = 10 км. Ответ запишите в м/с.

Решение:

Оценим массу нашего астероида. (Для справки, его размер примерно соответствует размеру Фобоса - спутника Марса. Масса Фобоса больше примерно в 2,5 раза больше, но наш астероид ледяной). $M_{astr} = \frac{4}{3}\pi\rho_{ast}R_{astr}^3 \approx 3{,}8\cdot 10^{15}$ кг.

Для того, чтобы космонавт мог находится на желаемой орбите, у него должна быть скорость, квадрат которой равен $V^2 = \frac{GM_{astr}}{R_{astr}+H}$, при этом, чтобы подняться на нее, ему нужно изменить свою потенциальную энергию на величину $\Delta E_{pt} = -GM_{astr}\left(\frac{1}{R_{astr}+H} - \frac{1}{R_{astr}}\right)$, тогда полная энергия космонавта на орбите $E = \frac{GM_{astr}m}{2(R_{astr}+H)} - GM_{astr}m\left(\frac{1}{R_{astr}+H} - \frac{1}{R_{astr}}\right) = GM_{astr}m\left(\frac{1}{R_{astr}} - \frac{1}{2(R_{astr}+H)}\right) \approx 1{,}5$ кДж для человека средней массы.

Необходимая скорость может быть оценена из закона сохранения энергии (масса человека на расчет не влияет):

$$V = \sqrt{2GM_{astr}\left(\frac{1}{R_{astr}} - \frac{1}{2(R_{astr}+H)}\right)} \approx 6{,}1 м/с$$

Оценим теперь с какой скоростью он может прыгнуть. Для этого предположим, что космонавт сравнительно обычный человек и легко прыгает на 0,5м. Тогда минимальная скорость, которую он должен иметь на старте прыжка на Земле равна:

$V_0 = \sqrt{2 \cdot g \cdot H} \approx 3{,}2 м/с$, видно, что скорости которой может достичь космонавт, недостаточно.

Ответ:

Необходимая начальная скорость порядка 6,1м/с.

Задача 1.4.3 (2 балла)

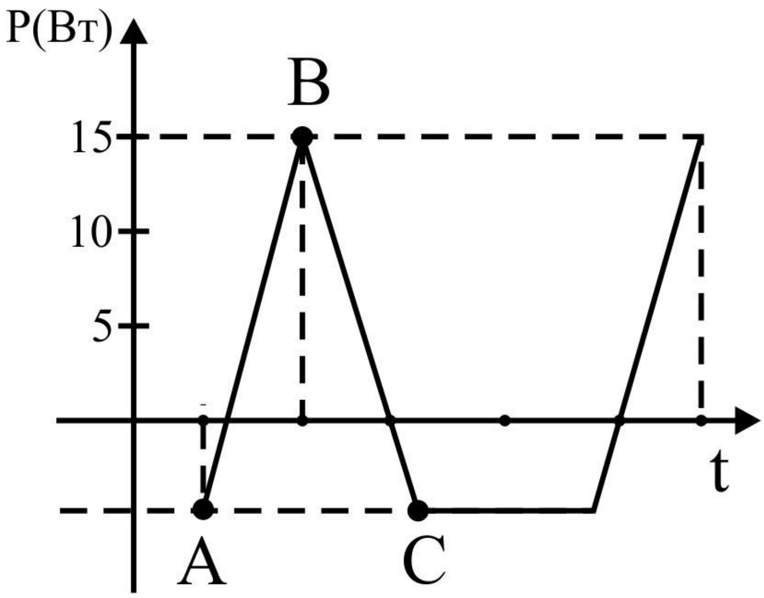
Условие:

Энергосистема спутника состоит из постоянной полезной нагрузки, солнечной батареи и аккумулятора. Информация о зависимости от времени потребляемой и вырабатываемой мощности за один оборот спутника представлена в таблице. Оцените по этой информации, какое количество энергии выработала солнечная батарея за один оборот спутника. Период обращения этого спутника 96 минут. Ответ дайте в кДж.

Время	Мощность зарядки (+) и разрядки (-) аккумулятора	Время	Мощность зарядки (+) и разрядки (-) аккумулятора
9:00	-5Вт	10:00	-5Вт
9:06	-5Вт	10:06	-5Вт
9:12	0Вт	10:12	-5Вт
9:18	5Вт	10:18	-5Вт
9:24	10Вт	10:24	-5Вт
9:30	15Вт	10:30	-5Вт
9:36	10Вт	10:36	-5Вт
9:42	5Вт	10:42	-5Вт
9:48	0Вт	10:48	0Вт
9:54	-5Вт	10:52	5Вт

Решение:

Проще всего решить эту задачу, превратив табличные значения в график. Тогда решение становится практически очевидным. Понятно, что 5 Вт, это потребление энергии постоянной нагрузкой, а выработка энергии солнечной батареей происходит сначала с линейным ростом мощности, а потом с таким же линейным спадом. Так же видно, что энергия вырабатывается батареей с 9:06 до 9:54, т.е. половину периода обращения.



Для того, чтобы найти полную выработанную энергию, достаточно найти площадь под графиком, т.е. площадь треугольника АВС. Нужно только учесть, что весь график сдвинут на «постоянное потребление», таким образом искомая энергия равна: 20Вт*24*60с/2 = 14,4кДж.

Ответ:
14,4 кДж.

Задача 1.4.4 (2 балла)
Условие:
Уже сейчас околоземная орбита сильно загрязнена и ситуация с каждым годом не становится лучше. Оцените, какова вероятность встречи спутника с опасным космическим мусором за один оборот вокруг Земли, если бы вокруг Земли оказался 1 мусорный объект на 1км^3. Максимальная площадь поверхности спутника 10 м^2, а период его обращения по круговой орбите вокруг Земли 90 минут. *Для оценки можно считать, что мусор в пространстве распределен равномерно.*

Решение:
Найдем «длину свободного пробега» такого спутника. Оценить ее можно, с учетом данной по условию равномерности распределения мусора, считая, что $\eta S l_{кр} = 1$, где η – плотность космического мусора, а S – площадь спутника.
Оценим скорость спутника, зная его период обращения:

$$\begin{cases} \dfrac{V^2}{R} = \dfrac{gR_з^2}{R^2} \\ \dfrac{2\pi R}{T} = V \end{cases} \Rightarrow R = \sqrt[3]{\frac{gR_з^2 T^2}{4\pi^2}}$$

Вероятность встретить мусор равна отношению расстояния, пройденного спутником за один оборот, к «длине свободного пробега.

$$\chi = \frac{2\pi R}{l_{кр}} = 2\pi R\eta S = 2\pi\eta S\sqrt[3]{\frac{gR_з^2 T^2}{4\pi^2}} = \eta S\sqrt[3]{2\pi g R_з^2 T^2} = \quad 42\%$$

Ответ:
(42 $\pm$ 15)%

Задача 1.4.5 (2 балла)
Условие:
Одним из перспективных двигателей для космических аппаратов является солнечный парус. Это очень легкий и хорошо отражающий свет от Солнца материал, при отражении от которого свет сообщает парусу импульс. Перспективные материалы для солнечного паруса должны иметь плотность 0,1г/м3 при коэффициенте отражения 99.9% Оцените, какое максимальное ускорение может получить космический аппарат массой (без паруса) 1кг с помощью такого паруса на расстоянии равном расстоянию от Земли до Солнца. Мощность солнечного света попадающего на единицу поверхности на этом расстоянии (солнечная постоянная) равна 1387Вт/м2. Ответ выразите в м/с2.

Свет переносит импульс пропорциональный своей энергии: $p = \dfrac{E}{c}$, *где E - энергия света, а c - скорость света.*

Решение:

Импульс переданный аппарату можно записать, как $\Delta p = 2\alpha \frac{E}{c} + (1-\alpha)\frac{E}{c}$, из-за того, что часть света отразится, а часть поглотится.

Тогда сила с которой свет действует на парус может быть выражена как $F = \frac{\Delta p}{\Delta t} = (1+\alpha)\frac{KS\Delta t}{\Delta t} = (1+\alpha)KS\Delta t$, где K - солнечная постоянная.

Ускорение аппарата равно $a = \frac{F}{m}$, масса аппарата складывается из массы основной части и массы паруса: $m = m_0 + \sigma S$, где σ- поверхностная плотность материала паруса.

Тогда ускорение равно: $a = \frac{(1+\alpha)KS\Delta t}{m_0 + \sigma S} = \frac{(1+\alpha)K\Delta t}{\frac{m_0}{S} + \sigma}$. Видно, что эта величина при S - стремящемся к бесконечности стремится к $a_{кр}$ = 0,091м/с^2.

Ответ:
0,091м/с^2

1.5 Первая попытка
Задачи по информатике (9 класс)

Задача 1.5.1 (1 балл)
Условие:

Известно, что вторая космическая скорость -- это наименьшая скорость, необходимая объекту для преодоления гравитационного притяжения небесного тела и покидания замкнутой орбиты вокруг него. В этой задаче вам необходимо вычислить вторую космическую скорость для разных космических тел. Каждый тест, на котором задача будет проверяться, будет состоять из двух чисел M -- масса космического тела в кг $\cdot\ 10^{21}$ и r — его радиус в м. Высоту тела над поверхностью планеты h считать равной 0 м. В этой задаче необходимо принять значения гравитационной постоянной равной $6,674 \cdot 10^{-11}$м$^3\cdot$с$^{-2}\cdot$кг$^{-1}$

Формат входных данных:

В первой строке дано целое положительное число $0 < M \le 10^9$ -- масса космического тела, выраженная в килограммах. Во второй строке дано целое положительное число $0 < r \le 10^9$ -- радиус тела, выраженный в метрах.

Формат выходных данных:

В единственной строке выведите одно число, являющееся второй космической скоростью для данного космического тела, выраженной в км/с. При необходимости округлите результат до двух знаков после запятой.

Решение:

Необходимо воспользоваться формулой второй космической скорости. Для того, чтобы результат поместился в переменную можно сократить степени 10 заранее.

```
def solve():
    M, R = map(int, input().split())
    G = 6.674
    return round(10**2* (2*G*M/R)**0.5, 2)
```

Задача 1.5.2 (2 балла)

Условие:

При отборе сотрудников на любую работу практически всегда есть конкуренция. Отбор космонавтов из кандидатов исключением не является. Только конкуренция слишком высока, а значит и критерии выбора людей должны быть очень четкими и понятными. Конкурсная комиссия общается с претендентами и выясняет множество подробностей о состоянии здоровья, образовании, навыках коммуникации. Перед вами стоит задача провести упрощенный отбор космонавтов для полета на орбиту.

Формат входных данных:

В первой строке дано число $0 < N \leq 10^5$ -- количество претендентов и число M -- количество людей, которых необходимо выбрать. Далее в N строках следует по 3 числа каждое, из которых показывает одну из характеристик космонавта выраженную числом от 0 до 1000. Где 1000 обозначает отличный результат, а 0 крайне неудовлетворительный.

Формат выходных данных:

Требуется выбрать и вывести на экран список из сумм баллов M лучших претендентов, отсортированный по результатам оценок конкурсной комиссии. Кандидаты сортируются по сумме своих оценок. Если претендентов меньше, чем количество людей, которых необходимо отобрать, выведите вместо результата недостающего претендента нуль -- результат для наихудшего возможного прохождения теста.

Решение:

Так как нам не надо восстанавливать разбалловку по характеристикам, при чтении будем сразу записывать в массив суммы. Далее необходимо отсортировать массив по убыванию любой удобной для вас сортировкой. Если m > n, то необходимо добавить в массив недостающее количество нулей.

```
def solve():
    res = list()
    n, m = map(int, input().split())
    for _ in range(n):
        x, y, z = map(int, input().split())
        res.append(x + y + z)
    res = sorted(res, reverse=True)
    for i in range(max(0, m - n)):
        res.append(0)
    ans = list()
    for i in range(m):
        ans.append("{}\n".format(res[i]))
    return ''.join(ans)
```

Задача 1.5.3 (2 балла)

Условие:

При подготовке документации по очередному космическому проекту было обнаружено, что часть модулей зависит от других модулей системы и, что самое ужасное, никто особо не помнит какие модули от каких зависят. В описании, конечно, хочется расположить модули таким образом, чтобы сначала прочитать об основных модулях, а потом о тех, которые на них опираются. Мы не просим вас расположить модули для документации в правильном порядке, а просим лишь ответить возможно ли это сделать или найдется набор модулей, которые опираются друг на друга по кругу.

Формат входных данных:

В первой строке дано число модулей в системе $3 < N \leq 10^5$ и число описанных зависимостей $M \leq 10^7$. В следующих M строках записаны пары $1 \leq s_i f_i \leq N$. Первое число -- это номер зависимого модуля, второе число -- номер базового модуля, на который опирается данный зависимый. Некоторые модули могут быть связаны больше чем одной зависимостью. В этом случае они будут указаны больше одного раза в списке.

Формат выходных данных:

В единственной строке выведите ответ на вопрос можно ли расположить модули в документации в порядке от базовых подмодулей к составным модулям так, чтоб к моменту прочтения любой главы документации все используемые в ней подмодули были описаны в предыдущих главах. Если возможно, выведите на экран "Yes", иначе “No"

Решение:

Модули системы можно представить как вершины графа, а упоминания в документации по модулю A модуля B представить как ориентированное ребро из A в B. Считаем граф как список ребер. Далее для поиска циклов будем использовать алгоритм поиска в глубину(dfs). Будем раскрашивать вершины в 3 цвета: изначально все вершины помечены 0, что означает, что ни одна из них не посещена. Дальше для каждой не посещенной вершины мы будем вызывать процедуру поиска в глубину.

Работает она так:

вершина, от которой стартует поиск в глубину помечается 1. Это означает, что алгоритм побывал в данной вершине. Для всех вершин до которых можно добраться из стартовой возможно два варианта: либо в них мы уже побывали(а значит, они помечены 1) и значит, что можно найти цикл в графе, либо мы в вершине ещё не посещена и мы стартуем dfs из нее. Когда мы обошли все вершины доступные из стартовой, мы помечаем стартовую вершину цветом 2. Это позволит нам не спутать ситуации на картинке

```
def dfs(v, visited, module):
    visited[v] = 1
    for n_ver in module[v]:
        if visited[n_ver] == 0:
            if dfs(n_ver, visited, module):
                return True
        elif visited[n_ver] == 1:
            return True
    visited[v] = 2
    return False

def solve():
    N, M = map(int, input().split())
    module = []
    for i in range(N):
        module.append(list())
    for i in range(M):
        s, f = map(int, input().split())
        module[s-1].append(f-1)
    visited = [0] * N
```

```
    for vertex_id in range(N):
        if visited[vertex_id] == 0:
            res = dfs(vertex_id, visited, module)
            if res:
                return "No"
    return "Yes"
```

Задача 1.5.4 (3 балла)

Условие:

Построение карт небесных тел интересовало людей уже достаточно давно. Атласы звездного неба видели многие. Уже тысячелетиями люди объединяют звезды в созвездия, проводя воображаемы линии между видимыми звездами. Но задумывались ли вы когда-нибудь о реальной длине таких нереальный линий? Введем неподвижную систему координат, на которой установим направление трех перпендикулярных осей координат Ox, Oy, Oz.

Формат входных данных:

В первой строке задано число звезд $0 < N \leq 10^5$ в созвездии и число $0 < M \leq 10^7$ линий, соединяющих созвездия. В следующих N строках описаны тройки чисел: целочисленные координаты по осям Ox, Oy, Oz. Далее в M строках записаны пары в которых числа: номера звезд в списке, соединенных воображаемой прямой. Нумерация звезд начинается с единицы. Будьте внимательны: если созвездие дважды включает какую-то линию, её необходимо учесть дважды.

Формат выходных данных:

Необходимо вывести на экран суммарную длину линий, соединяющих звезды в созвездия, округленную до двух знаков после запятой.

Решение:

Считываем координаты всех звезд в отдельный массив. Пока считываем ребра сразу считаем длину ребра и добавляем её к итоговому ответу. Длина ребра считается по формуле расстояния в пространстве $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$. Запоминать ребра не надо.

```
def solve(dataset):
    N, M = map(int, input().split())
    stars = list()
    ans = 0
    for i in range(N):
        x, y, z = map(int, input().split())
        stars.append((x, y, z))
    for i in range(M):
        f, s = map(int, data[1+N+i].split())
        dest = ((stars[f-1][0]-stars[s-1][0])**2 +
                (stars[f-1][1]-stars[s-1][1])**2 +
                (stars[f-1][2]-stars[s-1][2])**2)**0.5
        ans += dest
    return str(round(ans, 2))
```

1.6 Первая попытка
Задачи по информатике (10 - 11 класс)

Задача 1.6.1 (1 балл)

Условие:

Известно, что вторая космическая скорость -- это наименьшая скорость, необходимая объекту для преодоления гравитационного притяжения небесного тела и покидания замкнутой орбиты вокруг него. В этой задаче вам необходимо вычислить вторую космическую скорость для разных космических тел. Каждый тест, на котором задача будет проверяться, будет состоять из двух чисел M -- масса космического тела в кг $\cdot\ 10^{21}$ и r — его радиус в м. Высоту тела над поверхностью планеты h считать равной 0 м. В этой задаче необходимо принять значения гравитационной постоянной равной $6,674 \cdot 10^{-11}$м$^3 \cdot$с$^{-2} \cdot$кг$^{-1}$

Формат входных данных:

В первой строке дано целое положительное число $0 < M \leq 10^9$ -- масса космического тела, выраженная в килограммах. Во второй строке дано целое положительное число $0 < r \leq 10^9$ -- радиус тела, выраженный в метрах.

Формат выходных данных:

В единственной строке вывести одно число, являющееся второй космической скоростью для данного космического тела, выраженной в км/с. При необходимости округлите результат до двух знаков после запятой.

Решение:

Необходимо воспользоваться формулой второй космической скорости. Для того чтоб результат поместился в переменную и не произошло переполнения, можно сократить степени 10 заранее. В результате получим следующую формулу $10^2 * \frac{\sqrt{2GM}}{R}$

```
def solve():
    M, R = map(int, input().split())
    G = 6.674
    return round(10**2* (2*G*M/R)**0.5, 2)
```

Задача 1.6.2 (2 балла)

Условие:

У космического аппарата есть набор пронумерованных элементов оборудования(манипуляторов и датчиков). Каждый элемент оборудования находится в одном из четырех состояний: функционирует исправно, сломан, выключен или состояние неизвестно. Состояние оборудования передается на Землю в виде десятичного числа N, которое получится если выписать состояния манипуляторов и датчиков в порядке возрастания номеров. По полученному на Земле числу N определите минимально возможное число приборов на космическом аппарате. В случае если i-ый прибор сломан, на i-ое место записывается 0 В случае если i-ый прибор функционирует исправно, на i-ое место записывается 1 В случае если i-ый прибор выключен принудительно, на i-ое место записывается 2 В случае если i-ый прибор не сообщает свое состояние, на i-ое место записывается 3 После того как получена запись показаний, результат переводится в десятичное число N.

Формат входных данных:

На вход программы подается единственное целое неотрицательное число $N \leq 10^9$

Формат выходных данных:

В результате выполнения программы должно быть выведено на экран единственное число -- минимальное количество элементов оборудования на космическом аппарате, такое, что существует состояние оборудования для которого будет корректно описание числом N по приведенной инструкции.

Решение:

Записав показания манипуляторов, мы всегда получаем число в 4-системе счисления. Числа в 4-системе счисления легко переводятся. Для того чтоб узнать минимальное число, необходимо взять логарифм, например логарифм по основанию 4. Либо, если в вашем языке программирования проще взять логарифм по основанию 2, то результат логарифмирования необходимо разделить на 2. Это решение основывается на том, что логарифмирование даст нам максимальный, неравный 0 разряд. Логарифмировать надо по основанию системы счисления.

```
def solve():
    N = int(input())
    if N > 0:
        ans = math.ceil(math.log(N, 2)/2)
    else:
        ans = 0
    return ans
```

Задача 1.6.3 (3 балла)

Условие:

При отборе сотрудников на любую работу практически всегда есть конкуренция. Отбор космонавтов из кандидатов исключением не является. Только конкуренция слишком высока, а значит и критерии выбора людей должны быть очень четкими и понятными. Конкурсная комиссия общается с претендентами и выясняет множество подробностей о состоянии здоровья, образовании, навыках коммуникации. Перед вами стоит задача провести упрощенный отбор космонавтов для полета на орбиту.Каждый кандидат оценивается по 3 характеристикам. Значение каждой характеристики выражено числом от 0 до 1000. Характеристики в строке отсортированы по важности от более важной к менее важной.Вам потребуется выбрать и вывести на экран список из M лучших претендентов, отсортированный по результатам оценок конкурсной комиссии. При этом сравнивая претендентов сначала сравниваются более важные характеристики, потом менее важные.

Формат входных данных:

В первой строке дано число $0 < N \le 10^5$ -- количество претендентов и число $0 < M \le 10^5$ -- количество людей, которых необходимо выбрать.Далее в N строках следует по 3 числа через пробел каждое, из которых показывает одну из характеристик космонавта выраженную числом от 0 до 1000. Где 1000 обозначает отличный результат, а 0 -- крайне неудовлетворительный. На первом месте идет самая важная характеристика, на втором менее важная, на третьем менее важная, чем вторая характеристика.

Формат выходных данных:

Выведите M строк. В i-ой строке должны быть записаны через пробел значения трех характеристики i-ого кандидата в отсортированном списке претендентов. Если претендентов меньше, чем количество людей, которых необходимо отобрать, выведите вместо результата недостающего претендента три нуля -- результат для наихудшего возможного прохождения теста.

Решение:

В задаче необходимо выполнить сортировку по 3 параметрам. В языке python списки и кортежи сортируются по элементно. При равенстве первых элементов, сравниваются вторые элементы, при равенстве и первых и вторых элементов сравниваются третьи элементы.

В других языках можно написать любую сортировку вручную написав сравнение двух космонавтов.

```
def solve():
    res = list()
    n, m = map(int, input.split())
    for _ in range(n):
        x, y, z = map(int,input().split())
        res.append([x, y, z])
    for i in range(max(0, m - n)):
        res.append([0, 0, 0])
    res = sorted(res, reverse=True)

    ans = list()
    for i in range(m):
          ans.append("{} {} {}\n".format(res[i][0], res[i][1],
res[i][2]))
    return ''.join(ans)
```

Задача 1.6.4 (4 балла)

Условие:

Очередь - одна из самых естественных структур в мире. С очередями сталкиваются люди, желающие оплатить свои покупки в магазине и запросы, ожидающие обработки на серверах. Наша задача как раз про запросы. Космическая станция вынуждена передавать множество данных. Малая часть из них будет обработана уже на станции, однако большая часть будет передана на Землю и обсчитана специальными программами. Известно, что данные можно передавать наборами от одного пакета цифровой информации до трех. Скорость обработки набора значительно зависит от первого пакета цифровой информации в нем и незначительно от последующих.

Формат входных данных:

В первой строке дано единственное число $N \le 5000$ -- количество пакетов цифровой информации, передаваемых на Землю. Далее в N строках идет 3 числа. Первое число обозначает время в секундах для обработки набора состоящего только из этого пакета. Второе число -- время обработки набора, состоящего из текущего пакета и одного следующего за ним пакета. Третье число -- время обработки набора, который состоит из текущего и двух следующих за ним пакетов цифровой информации. Обработка каждого набора данных не превышает 1 часа.

Формат выходных данных:

Необходимо вывести единственное число -- минимальное время в секундах, за которое все пакеты информации будут обработаны на Земле.

Решение:

Для решения задачи для большого N сведем задачу к меньшему числу пакетов.
Для $N = 1$ ответ очевиден, так как существует единственный вариант отправить пакет.

Для N = 2 ответ будет равен минимальному значению из

1. скорости обработки первого со вторым пакетом вместе
2. суммы скоростей обработки по отдельности первого и второго пакета

Для N = 3 ответ будет равен минимальному значению из

1. скорости обработки трех пакетов вместе
2. суммы минимальной скорости обработки двух первых пакетов и скорости обработки третьего пакета отдельно
3. суммы скоростей обработки первого пакета и минимальной скорости обработки двух следующих пакетов, которая вычисляется также как ответ для варианта где N = 2

Можно продолжить рассматривать варианты для всё увеличивающихся N. Единственное, что необходимо понять, что ответ для N > 3 всегда можно получить опираясь на данные о скоростях обработки текущего объекта, и 2 предыдущих, а также зная оптимальное время для N’ = N - 1, N’’ = N - 2. Решение задачи опираясь на задачу с меньшими входными данными, последовательно расширяя объем данных называется динамическим программированием.

```
def solve():
    n = int(input())
    s = [0]
    for _ in range(n):
        s.append(list(map(int, input().split())))

    if n != 1:
        dp = [0] * (n + 1)
        dp[0] = 0
        dp[1] = s[1][0]
        dp[2] = min(s[1][0]+ s[2][0], s[1][1])

        for j in range(3, n + 1):
             dp[j] = min(dp[j - 1] + s[j][0], dp[j - 2] + s[j -
1][1], dp[j - 3] + s[j - 2][2])
        return dp[-1]
    else:
        return s[1][0]
```

1.7 Вторая попытка
Задачи по математике (9 класс)

Указание: *Во всех задачах, где это необходимо, ответ можно округлять до сотых, число π считать приближенно равным 3,14.*

Задача 1.7.1 (2 балла)

Условие:

Какое наименьшее значение может принимать сумма всевозможных попарных произведений $n=2k$ чисел, каждое из которых по модулю не превосходит единицу? (k – натуральное число)

Решение:

Перепишем искомую сумму чисел $x_1 \ldots x_n$ в виде

$$S = \frac{1}{2}((x_1 + \ldots + x_n)^2 - x_1^2 - \ldots - x_n^2) \geq \frac{1}{2}(-x_1^2 - \ldots - x_n^2) \geq -\frac{n}{2} = -k$$

Значение $S = -k$ достигается, например, при $x_1 = \ldots = x_k = 1, x_{k+1} = \ldots = x_n = -1$

Ответ:

$-k$

Задача 1.7.2 (2 балла)

Условие:

Пусть BD и CE - высоты остроугольного треугольника ABC. На прямую DE опущен перпендикуляр BF.

Найти длину EF, если $BC=b$, $cos(\angle ABC) = a$, $cos(\angle ACB) = c$.

Решение:

Пусть $\angle ACE = \angle ABD = \gamma$ (из подобия треугольников ACE и ABD). Тогда $AE=AC \cdot sin\,\gamma$, $AD=AB \cdot sin\,\gamma$, следовательно, треугольники ABC и ADE подобны и $\angle AED = \angle ACB$. Из треугольника BCE: $EB=BC \cdot cos\,\angle ABC = b \cdot a$.

Из треугольника BEF: $EF=EB \cdot cos\,\angle BEF = EB \cdot cos\,\angle AED = EB \cdot c = b \cdot a \cdot c$

Ответ:

$a \cdot b \cdot c$

Задача 1.7.3 (2 балла)

Условие:

В научном центре работают n исследователей, которые в течение года выезжали в различные экспедиции. В каждую могло поехать не более половины сотрудников. При этом в конце года оказалось, что любые два из них хотя бы один раз ездили вместе.

При каком наименьшем количестве экспедиций такое возможно? ($n=4k$, k – натуральное число)

Решение:

Каждый сотрудник за одну поездку мог пересечься с не более чем $(2k-1)$ коллегами, значит каждый выезжал не менее 3 раз (т.к. $2(2k-1)<(4k-1)$). Следовательно, экспедиций было не менее, чем $\frac{3 \cdot 4k}{2k} = 6$. Такая ситуация возможна, если разбить всех на 4 группы по k человек и отправлять в экспедиции по 2 группы во всевозможных комбинациях: 1 и 2, 1 и 3, … 3 и 4.

Ответ:
6

Задача 1.7.4 (2 балла)
Условие:
Два спутника (размерами которых можно пренебречь) летают вокруг планеты по одной круговой орбите в одном направлении. Первый совершает полный оборот на 0.5 часа быстрее, чем второй, при этом они встречаются каждый час, и ровно один раз. Найти скорость первого спутника (в км/ч), если длина орбиты равна S км.

Решение:
Пусть v – скорость первого спутника и u – скорость второго. Имеем $\frac{S}{v} = \frac{S}{u} - \frac{1}{2}$. Из условия, что за 1 час первый спутник делает ровно на 1 оборот больше, получаем $1 \cdot v - 1 \cdot u = S$. Выразив отсюда u и подставив в первое уравнение, получаем $v^2 - v \cdot S - 2S^2 = 0$. Решив это уравнение с учетом того, что $v \geq 0$, получаем $v = 2S$.

Ответ:
$2 \cdot S$

Задача 1.7.5 (2 балла)
Условие:
Найти наименьшее значение параметра a, при котором сумма кубов корней уравнения $(2a^2+3a+4)(x^2+x)+10a^3+27a^2+38a+24=0$ равна b.

Решение:
Пусть x_1, x_2 – корни уравнения. По теореме Виета $x_1 + x_2 = -1$,
$x_1 \cdot x_2 = \frac{10a^3+27a^2+38a+24}{2a^2+3a+4} = 5a + 6$
$x_1^3 + x_2^3 = (x_1 + x_2)(x_1^2 - x_1x_2 + x_2^2) = (x_1 + x_2)((x_1 + x_2)^2 - 3x_1x_2) = b$
Подставим: $b = -1(1 - 3(5a + 6))$, откуда $a = \frac{b-17}{15}$

Ответ:
(b-17)/15

1.8 Вторая попытка
Задачи по математике (10 - 11 класс)

Указание: *Во всех задачах, где это необходимо, ответ можно округлять до сотых, число π считать приближенно равным 3,14.*

Задача 1.8.1 (1 балл)
Условие:
Найти число решений в натуральных числах уравнения $x^2+2016=y^2$

Решение:
Перепишем уравнение в виде: $(y+x)(y-x)=2016$. Задача сводится к поиску количества вариантов представления числа *2016* в виде произведения двух целых чисел, где $a>b$ (т.к. x положителен), и $b>0$ (т.к. $a>0$ и $2016>0$), то есть числа a и b натуральны.

Тогда получим $2y=a+b$, $2x=a-b$, откуда видно, что числа a и b должны быть одинаковой четности.

Разложим число *2016* на простые множители: $2016=2^5 3^2 7$. Отсюда видно, что одно из чисел a и b обязательно четно, а значит и второе тоже четно. Остается разбить $2^3 3^2 7$ на два множителя (они обязательно будут различны, т.к. *7* входит в состав числа *2016* в первой степени). Это можно сделать $\frac{4\cdot3\cdot2}{2} = 12$ способами.

Ответ:
12

Задача 1.8.2 (2 балла)
Условие:

В прямоугольнике *ABCD* на диагональ *AC* опущен перпендикуляр *BE*. Пусть *F* – середина *AE*, *M* – середина *CD*. Найдите $cos\angle BFM$, если длина диагонали прямоугольника равна *d*, а *AF/AC=1/9*.

Решение:

Решим задачу векторным способом. Обозначим $\vec{BA} = \vec{a}$, $\vec{BC} = \vec{b}$, $\vec{AC} = \vec{c}$, тогда $\vec{b} = \vec{a} + \vec{c}$. $\vec{AF} = \vec{FE} = \alpha\vec{AC}$ (у нас по условию $\alpha = \frac{1}{9}$). Так как $(\vec{a}, \vec{b}) = 0$, то $(\vec{a}, \vec{a} + \vec{c}) = 0$, то есть $(\vec{a}, \vec{a}) =- (\vec{a}, \vec{c})$. Так как $\vec{BE} = \vec{a} + 2\alpha\vec{c}$, то $(\vec{a} + 2\alpha\vec{c}, \vec{c}) = 0$, то есть $(\vec{a}, \vec{c}) =- 2\alpha(\vec{c}, \vec{c})$, а $(\vec{a}, \vec{a}) = 2\alpha(\vec{c}, \vec{c})$.

Далее, $\vec{FM} = (1 - \alpha)\vec{c} + \frac{1}{2}\vec{a}$, $\vec{BF} = \vec{a} + \alpha\vec{c}$.

Тогда $(\vec{BF}, \vec{FM}) = (\vec{a} + \alpha\vec{c}, (1 - \alpha)\vec{c} + \frac{1}{2}\vec{a}) = (1 - \frac{\alpha}{2})(\vec{a}, \vec{c}) + \frac{1}{2}(\vec{a}, \vec{a}) + \alpha(1 - \alpha)(\vec{c}, \vec{c}) =$
$= (1 - \frac{\alpha}{2})(- 2\alpha)(\vec{c}, \vec{c}) + \frac{1}{2}\cdot 2\alpha(\vec{c}, \vec{c}) + \alpha(1 - \alpha)(\vec{c}, \vec{c}) = 0$

Следовательно, $cos\angle BFM = 0$.

Примечание: Как видим, в данной задаче ответ не зависит от длин или их отношений.

Ответ:
0

Задача 1.8.3 (2 балла)
Условие:

В научной лаборатории работают ученые и программисты, при этом женщин работает на 100 меньше, чем мужчин. Сколько всего сотрудников в лаборатории, если известно, что среди женщин ученых на 20 больше, чем программистов; что среди ученых мужчин в 3 раза больше, чем женщин; а среди программистов – мужчин больше, чем женщин в n раз.

Решение:

Обозначим количество женщин–ученых через x. Тогда женщин-программистов работает *x-20,* мужчин-ученых $3x$, а мужчин-программистов *n(x-20).* Из первого условия имеем: $2x - 20 + 100 = 3x + n(x - 20)$, откуда $x = 20 + \frac{60}{n+1}$. Общее число сотрудников равно $x + x + 20 + 3x + n(x - 20) = x(n + 5) - 20(n + 1)$. Подставив сюда x, получим $80 + \frac{60(n+5)}{n+1}$.

Ответ:
80+(60*(n+5)/(n+1))

Задача 1.8.4 (2 балла)

Условие:

Космонавт шел по поверхности планеты сферической формы в одном направлении. В какой-то момент оказалось, что, чтобы увидеть точку начала пути, ему нужно подняться вертикально вверх над поверхностью планеты на высоту R.

Какое расстояние (длину пути по поверхности) прошел космонавт, если радиус планеты равен R.

Решение:

Пусть O – центр сферы (планеты), A - начало пути космонавта, B - конец пути. Пусть, подпрыгнув на высоту R, космонавт окажется в точке C (точки O, B и C лежат на одной прямой). Рассмотрим треугольник OAC. $OA=R$, $OC=2R$, $\angle OAC=\frac{\pi}{2}$,т.к. AC – касательная. Тогда $\cos \angle AOC= \frac{1}{2}$, т.е. $\angle AOC = \frac{\pi}{3}$ Следовательно, длина дуги $AB = \frac{\pi R}{3}$

.

Ответ:

$\frac{\pi R}{3}$

Задача 1.8.5 (2 балла)

Условие:

Числа a и b удовлетворяют равенствам: $a^3-3a^2+4a=n$, $b^3-3b^2+4b=4-n$.

Найти сумму $a+b$.

Решение:

Рассмотрим функцию $f(x) = x^3-3x^2+4x = (x-1)^3+(x-1)+2$.

Пусть $g(y) = y^3+y$. Это нечетная и строго возрастающая функция. Заметим, что $g(a)=f(a)-2=n-2$, $g(b)=f(b)-2=2-n$. Отсюда, в силу нечетности g, получаем $a-1=-(b-1)$, то есть $a+b=2$.

Ответ:

2

Задача 1.8.6 (2 балла)

Условие:

Сферический спутник движется по круговой орбите вокруг Солнца в два раза ближе чем Земля. Эффективный коэффициент отражения спутника: α = 75%. Оцените, какой дополнительный импульс необходимо гасить этому спутнику ежегодно из-за солнечного света. Ответ выразите в кг*м/с округлив до целых.

*Мощность солнечного света попадающего на единицу поверхности на этом расстоянии (солнечная постоянная) равна 1387Вт/м^2. Свет переносит импульс пропорциональный своей энергии: *, где Е - энергия света, а с - скорость света.*

Решение:

С учетом отражения и того, что свет от Солнца всегда действует по радиусу, соединяющему центр Солнца и спутник, можно записать.

*.   , где

энергия переданная спутнику за год:

*

Коэффициент 4 появляется из-за того, что спутник находится в 2 раза ближе, чем Земля и энергии получает в 4 раза больше.

Тогда итоговое приращения импульса равно 3205 кг · м/с.

Ответ:
$\Delta p = 3205$ кг · м/с

Задача 1.8.7 (2 балла)
Условие:
В таблице представлена зависимость скорости расхода энергии, запасенной в аккумуляторе спутника от времени. Известно, что на всех этапах работы, для которых приведена таблица, эта скорость была постоянной или зависела от времени линейно. Найдите разницу между возможным максимальным и минимальным расходом энергии за время с 0:30 до 1:50. Ответ дайте в джоулях с точностью до целых.

Время (м:с)	Мощность (Вт)
0:10	100
0:20	100
0:30	100
0:40	80
0:50	75
1:00	70
1:10	65
1:20	60
1:30	55
1:40	50
1:50	30
2:00	30

Решение:
Поскольку неизвестно как именно зависела мощность от времени, для оценки минимального и максимального значения можно предположить, что на каждом промежутке между известными значениями для максимального расхода энергии мощность оставалась постоянной и равной начальному значению на промежутке, а затем почти мгновенно падала до конечного значения, а для минимального расхода энергии - наоборот, сначала падала до конечного значения, а затем оставалась постоянной. Тогда можно оценить разницу выработанных работ, как (2*20*10+6*5*10)Дж = 700 Дж.

Ответ:
700 Дж

Задача 1.8.8 (2 балла)
Условие:
При подготовке полета на Венеру исследовались различные термостойкие материалы. В процессе изучения образцов одного из материалов в одной серии экспериментов было получено отношение удельных теплоемкостей материала в твердой и жидкой фазах α = 1,36. Другая серия экспериментов была посвящена изучению плавления образца. В ней была получена следующая зависимость температуры от времени: температура в течение времени τ = 10 мин, росла, затем в течении такого же времени оставалась неизменной, а затем еще столько же времени снова росла. Опыт моделировал нагревание первоначально твердого образца в горячей атмосфере Венеры. Полезная мощность нагревателя в расчете на единицу массы образца в течение всего опыта оставалась неизменной и равной P = 289Вт/кг. Начальная и конечная температуры в этом опыте T_1 = 20˚С, T_2 = 160˚С.
Определите по этим экспериментальным данным физические параметры

плавления материала (температуру плавления и удельную теплоту плавления. Ответ дайте в градусах цельсия с точностью до десятых и в кДж/кг с точностью до целых.

Решение:

На каждом этапе было передано одинаковое удельное количество теплоты:
$Q = P\tau$.
Тогда можно написать уравнение теплового баланса для каждого из трех этапов:
Нагревание твердого тела: $c_1(t_{пл} - 20℃) = P\tau$
Плавление: $\lambda = P\tau$
Нагревание жидкости: $c_2(160℃ - t_{пл}) = P\tau$
Из второго уравнения сразу получаем удельную теплоту плавления, которая равна (нужно не забыть перевести время в секунды) 173кДж/кг.
Разделив первое уравнение на третье и используя известное из условия отношение теплоемкостей получим искомую температуру плавления: $\frac{c_1(t_{пл} - 20℃)}{c_2(160℃ - t_{пл})} = 1$, откуда
$t_{пл} = 79℃$

Ответ:

Температура 79,3˚С, теплота плавления 173 кДж/кг.

Задача 1.8.9 (1 балл)

Условие:

Первый спутник Земли представлял собой металлический шар с антеннами, радиусом R = 0,58 м. Какого размера было отражение Солнца в Спутнике, во время первого облета Земли? Найдите диаметр отражения Солнца с в мм с точностью до десятых.

Диаметр Солнца 1,4 ·10⁶ км, а расстояние до него 150 ·10⁶ км?

Решение:

Спутник можно представить, как выпуклое сферическое зеркало с радиусом, равным радиусу спутника. Известно, что фокусное расстояние сферического зеркала равно половине его радиуса, тогда по формуле сферического зеркала увеличение равно *, где d - расстояние до Солнца. Тогда размер изображения *2,7мм, с учетом того, что расстояние до Солнца d много больше радиуса Спутника. (D - диаметр Солнца).

Ответ:

2.7мм.

1.9 Вторая попытка
Задачи по математике (11 класс)

Задача 1.9.1 (3 балла)

Условие:

Космический аппарат, запущенный на перехват спутника, движущегося по геостационарной орбите, оказался на орбите спутника в точке диаметрально противоположной его положению и начал движение навстречу с постоянной по модулю скоростью. Аппарат оснащен двигателями с силой тяги 1200Н,. Через 150 минут аппарат встретился со спутником. Какова масса аппарата?

За какое минимальное время космический аппарат догонит геостационарный спутник, не сходя с орбиты спутника и двигаясь против направлении его движения. До начала движения спутник и аппарат находятся в диаметрально противоположных точках орбиты. Аппарат оснащен двигателями с силой тяги 1017Н, масса аппарата 90кг. Ответ выразите в минутах.

Решение:

Чтобы продолжать двигаться по той же орбите с большей скоростью, аппарат должен своими двигателями создавать дополнительный вклад в центростремительное ускорение.

Тогда второй закон Ньютона может быть записан как:

$\frac{mV^2}{R} = \frac{mgR_з^2}{R^2} + F$, где R - расстояние между центром Земли и высотой геостационарной орбиты, V - скорость аппарата, m - его масса, F - сила тяги двигателей. Тогда наибольшая скорость, которую может развить аппарат:

$V = \sqrt{\frac{gR_з^2}{R} + \frac{FR}{m}}$, скорость спутника в свою очередь можно оценить зная (или вычислив) высоту геостационарной орбиты $U = \frac{2\pi R}{T}$, где $T = 24$ч.

Найдем скорость аппарата из соотношения:

$$V\tau + U\tau = \pi R$$

Тогда:

$$m = \frac{FR}{\left(\frac{\pi R}{\tau} - U\right)^2 - \frac{gR_з^2}{R}} = 401 \text{кг}$$

Ответ:

401+-1кг.

Задача 1.9.2 (2 балла)

Условие:

Космический зонд приземлился на небольшом сферическому астероиде из чистейшего водяного льда. Радиус астероида R = 10 км. Один из космонавтов, отправился на прогулку вокруг астероида и оказался в точке диаметрально противоположной месту посадки. За какое минимальное время он мог туда добраться на вездеходе, максимальная скорость которого по льду на Земле составляла 25км/ч? Ответ дайте целым числом минут. Вращением астероида пренебрегите.

Плотность льда 917кг/м^3.

Решение:

Несмотря на то, что решение задачи кажется очевидным, это очевидное решение неправильно. Вездеход не сможет развить свою максимальную скорость, т.к. при достижении первой космической скорости для астероида полностью потеряет сцепление с поверхностью и не сможет двигаться. Таким образом, необходимо проверить, величину первой космической скорости для астероида.

Она оказывается равной 5,06м/с =18,2 км/ч. Видно, что эта скорость ниже, чем максимальная скорость вездехода на Земле, таким образом, именно она ограничивает скорость вездехода на астероида.

Тогда искомое время равно *=103 минуты

Ответ:

103 минуты

Задача 1.9.3 (2 балла)

Условие:

В таблице представлена зависимость скорости расхода энергии, запасенной в аккумуляторе спутника от времени. Известно, что на всех этапах работы, для которых приведена таблица, эта скорость была постоянной или зависела от времени линейно. Найдите разницу между возможным максимальным и минимальным расходом энергии за время с 0:30 до 1:50. Ответ дайте в джоулях с точностью до целых.

0:10	100
0:20	100
0:30	100
0:40	140
0:50	150
1:00	160
1:10	170
1:20	180
1:30	190
1:40	200
1:50	220
2:00	220

Решение:

Поскольку неизвестно как именно зависела мощность от времени, для оценки минимального и максимального значения можно предположить, что на каждом промежутке между известными значениями для максимального расхода энергии мощность оставалась постоянной и равной начальному значению на промежутке, а затем почти мгновенно падала до конечного значения, а для минимального расхода энергии - наоборот, сначала падала до конечного значения, а затем оставалась постоянной. Тогда можно оценить разницу выработанных работ, как

(1*40*10+1*20*10+6*10*10)Дж = 1200Дж.

Ответ:
1200 Дж

Задача 1.9.4 (2 балла)
Условие:

Для уничтожения спутников были предложены, в частности, ракеты, создающие облако осколков, вытянутое вдоль орбиты, причем максимальное отдаление осколков от окружности орбиты составляет 20м. Оцените массу осколков, составляющих основу боеголовки такой ракеты, если она должна поразить спутник с 80% вероятностью при первом прохождении спутника через облако. Эффективная площадь поверхности спутника, против которого нацелена ракета 1 м^2. Масса одного осколка 10г. Ответ дайте в килограммах, с точностью до целых.

Решение:

$\eta Sl = 0.8$, где η – плотность космического мусора, а S – площадь спутника, а l - длина облака. Отсюда $\eta = \dfrac{0,8}{Sl}$

Число поражающих элементов равно $N = \eta V$, где V - объем облака, который равен $V = \pi R_{оск}^2 l$, тогда $N = \dfrac{0,8}{Sl}\pi R_{оск}^2 l = \dfrac{0,8\pi R_{оск}^2}{S} =$ 1001, тогда масса всех осколков 10кг.

Ответ:
10+-1кг

Задача 1.9.5 (1 балл)
Условие:

Юный астроном Василий, регулярно просматривает снимки с телескопа Хаббл, рассчитывая обнаружить на них изображение инопланетных кораблей. Оцените, каков минимальный размер объекта, находящегося на расстоянии 1 а.е. от Земли, теоретически доступный для обнаружения Василию. Укажите размер в км, с точностью до целых. Диаметр телескопа Хаббл 2.4 м. *1 а.е. - расстояние от Земли до Солнца.*

Угловой дифракционный предел телескопа может быть рассчитан как φ=1,22λ/D, где φ -минимальный угловой размер наблюдаемого объекта, λ - длина волны света, а D - диаметр объектива.

Решение:

Для космического телескопа единственным существенным ограничением на размер наблюдаемого объекта является дифракционный предел.

Угловой дифракционный предел телескопа может быть рассчитан как φ=1,22λ/D, где φ -минимальный угловой размер наблюдаемого объекта, λ - длина волны света, а D - диаметр объектива.

Для того, чтобы искомый размер оказался минимальным будем искать в синей области спектра, тогда $\lambda = 420$ нм. Искомый размер может быть вычислен как
$x = \frac{1.22\lambda}{D}L = 32$ км

Для красной области спектра размер объекта будет около 50км. Достаточно большая погрешность ответа вводится для учета границы спектра.

Ответ:
32 ± 3 км

1.10 Вторая попытка
Задачи по информатике (9 класс)

Задача 1.10.1 (1 балл)

Условие:

Часто, обсуждая космос, в особенности астрономию, необходимо понимать что такое орбита и как тела по ней движутся. В частности длина орбиты планет солнечной системы позволяет определить длину года. От вас требуется определить длину орбиты с погрешностью до 0.7%. Для решения задачи, считать π≈3.14. Ответ округлить до двух знаков после запятой.

Формат входных данных:

В этой задаче вам заданы два целочисленных параметра эллипса: 1≤a≤ $1 \le a \le 10^9$ — длина большой полуоси, $1 \le b \le 10^9$ — длина малой полуоси. Числа заданы в единственной строке и разделенных пробелом.

Формат выходных данных:

Выведите на экран единственное число -- длину орбиты с точностью до 2 знаков после запятой.

Решение:

Орбита имеет форму эллипса, о чем неявно сказано в условии. Для решения задачи воспользуемся формулой длины эллипса, изучаемой в школе. Важно понимать, что формула не является точной.

$$4\pi \frac{ab+(a-b)^2}{a+b}$$

```
a, b =  map(int, input().split())
l = 4.0 *(3.14 * a * b + (a - b)**2)/(a + b)
print(round(l, 2))
```

Задача 1.10.2 (3 балла)

Условие:

Космические корабли обшиты специальным материалом, в состав которого кроме всего прочего входят алюминий и титан. Казалось бы обшивка, состоящая из таких материалов, должна быть надежной. Но надежна она только до тех пор пока она целая. По этой причине при продолжительном использовании аппарата его обшивку необходимо регулярно проверять. Эту задачу может взять на себя специальный манипулятор.

Рассмотрим упрощенную модель: Каждый лист обшивки представляет собой круг на плоскости космического корабля. Манипулятор проверяет за раз несколько точек. В данной задаче манипулятору заданы несколько точек на одной окружности. Пример точек, которые могут быть заданы отмечены на рисунке зеленым.Кроме тех точек, которые манипуляторы заданы жестко, необходимо проверить дополнительно точку, которая находится в центре симметрии листа обшивки. Напишите программу, которая по входным данным определит координаты центра симметрии листа.

Формат входных данных:

В первой строке задано целое неотрицательное число k≤6 -- количество известных пар координат, заданных манипулятору на листе обшивки. В следующих k строках находится по 2 целых числа x,y разделенных пробелом. $-10^9 \le x,y \le 10^9$ -- координата очередной точки на окружности по оси x и по оси y соответственно.

Формат выходных данных:

В единственной строке выведите: Либо строку "Impossible" для случаев, когда однозначно определить центр симметрии листа невозможно.Либо два числа разделенные пробелом -- координаты x,yx,yx, y центра листа с точностью до 2 знаков после запятой.

Решение:

Для решения можно составить систему уравнений для трех точек на окружности. Уравнение окружности выглядит так:

$$x_0^2 + y_0^2 = r^2$$

При этом r -- расстояние от центра окружности до точки на окружности. То есть для каждой точки из входных данных верно, что

$$(x_i - x_0)^2 + (y_i - y_0)^2 = r^2$$

Выпишем 3 любые уравнения из входных данных:

$$x_0 + y_0 = (x_1 - x_0)^2 + (y_1 - y_0)^2$$
$$x_0 + y_0 = (x_2 - x_0)^2 + (y_2 - y_0)^2$$
$$x_0 + y_0 = (x_3 - x_0)^2 + (y_3 - y_0)^2$$

Далее находим решение уравнения в виде формулы.

Есть и второй способ. Возьмем 3 любые точки из входного множества и построим треугольник с вершинами в этих точках. Теперь нам необходимо найти координату центра описанной окружности, которая в свою очередь является точкой пересечения серединных перпендикуляров. Построим два любых серединных перпендикуляра и найдем точку их пересечения.

```
def solve(dataset):
    dataset = dataset.splitlines()
    k = int(dataset[0])
    if k < 3:
        return "Impossible"
    x1, y1 = map(float, input().split())
    x2, y2 = map(float, input().split())
    x3, y3 = map(float, input().split())
    A = x2 - x1
    B = y2 - y1
    C = x3 - x1
    D = y3 - y1
    E = A * (x1 + x2) + B * (y1 + y2)
    F = C * (x1 + x3) + D * (y1 + y3)
    G = 2 * (A * (y3 - y2) - B * (x3 - x2))

    Cx = round(((D * E - B * F) / G), 2)
    Cy = round(((A * F - C * E) / G), 2)
    return "{} {}".format(Cx, Cy)
```

Задача 1.10.3 (2 балла)

Условие:

В результате измерений ученые собирают достаточно большое количество данных. Их можно представлять по разному. Однако два наиболее часто используемых варианта: текстовое представление и графическое представление. В этом задании вашей задачей будет перевод информации из текстовой в графическую. В каждом тесте вам будет дан набор данных, по которым потребуется построить горизонтальную гистограмму.

Формат входных данных:

В первой строке дано целое число $1 \leq N \leq 10^4$ -- количество записей, которые необходимо обработать.В следующих N строках будет дано по паре название -- значение. Название это строка, которая не содержит пробелов. Значение -- целое число $0<x\leq 10$. Значение показывает на сколько увеличилось суммарное значение для данного названия. Название и значение разделены одним пробелом.

Формат выходных данных:

Выведите гистограмму по входным данным. Порядок столбцов гистограммы должен совпадать с порядком в котором впервые в данных встретились названия столбцы. В случае если на данной высоте в данной колонке гистограмма ещё не закончилась выведите "|". Если закончилась, выведите ".". Как минимум одна строка должна оканчивается на "|".

Разбор примера: если просуммировать все значения получим a = 6 d =2 v = 5 ! = 3 q = 4 e = 2. Отобразим в том порядке в котором названия впервые встретились.Замечание: Внимательно изучите примеры входных данных и результатов, которые должны быть получены. Если результат вашей программы на тестах из условия не похож на результат теста в примере, ваша программа гарантировано не пройдет тестирование.

Построить горизонтальную гистограмму.

Решение:

Заведем две структуры: во-первых, словарь, в котором будет храниться суммарное число значений для каждого символа. Помните, что словарь в языке python является неупорядоченным множеством. Во-вторых, заведем массив, в котором будем хранить порядок первого появления каждого символа.

Считываем в цикле значения, изменяем необходимые структуры данных. Параллельно находим максимальное суммарное значение по одному символу.

После того как считаны все данные и известен максимум в цикле выводим результат на экран. Порядок вывода берем из массива, значения берем из словаря по ключу, равному текущему значению в массиве.

```
def solve():
    n = int(input())
    tmp = dict()
    pos = list()
    max_col = 0
    res = list()
    for _ in range(n):
        key, value = input().split()
        if key in tmp:
            tmp[key] += int(value)
        else:
            tmp[key] = int(value)
            pos.append(key)
        max_col = max(max_col, tmp[key])
    for letter in pos:
        for i in range(max_col):
            if tmp[letter] > i:
                res.append('|')
            else:
```

```
                res.append('.')
            res.append('\n')
        return ''.join(res)
```

Задача 1.10.4 (4 балла)

Условие:

Настя и Лиза подруги по переписке. И иногда им очень хочется встретиться и обсудить свои хобби вживую, но расстояние является большой проблемой для школьниц. Поэтому они часто с надеждой изучают карту своей страны. В один из таких вечеров девочки задумались, а какое расстояние в среднем должен преодолеть опытный водитель, чтоб добраться из одного города в другой. Ваша задача помочь девочкам выяснить ответ на этот вопрос. На карте отмечены NNN городов и MMM дорог между ними. Для каждой дороги между городами известна её длина. Ваша задачи узнать кратчайшие пути между всеми парами городов и найти среднюю длину полученных путей.

Формат входных данных:

В первой строке заданы два натуральных числа $N \leq 1000, M \leq 10000$. В следующих M строках заданы тройки чисел -- два номера городов, которые соединены данной дорогой и длина данной дороги. Города пронумерованы числами от 1 до N. Длины дорог целые числа не превышающие 1000. Между парой городов может быть несколько дорог. Гарантируется, что из любого города на карте можно попасть в любой другой, однако не гарантируется, что есть прямую дорога между любой парой городов.

Формат выходных данных:

В единственной строке выведите среднюю длину кратчайших путей между всеми парами городов, округленную до 2 знаков после запятой.

Решение:

Дороги и города можно представить в виде графа. Тогда для решения задачи нам необходимо знать расстояние между каждой пары дорог. Один из самых эффективных алгоритмов для решения данной задачи -- алгоритм Флойда Уоршелла, подразумевающий что на k-ом шаге мы знаем правильные расстояния между всеми парами городов для графа, в котором остались только вершины с номерами от 0 до k-1.
На k-ом шаге мы пересчитываем расстояние между каждой парой вершин по формуле:
`d[i][j] = min(d[i][j], d[i][k] + d[k][j])`

Фактически мы проверяем не появился ли более выгодный путь между вершинами `i` и `j` после добавления в граф вершины номер k и всех ее ребер к уже включенным вершинам.

```
def solve(dataset):
    INF = 10**12
    data = dataset.splitlines()
    n, m = map(int , data[0].split())
    d = list()
    for sm in range(n):
        d.append([INF]*n)
    for tm in range(n):
        d[tm][tm] = 0

    for line in data[1:]:
        st, fn, wei = map(int, line.split())
        st -= 1
        fn -= 1
        d[st][fn] = min(d[st][fn], wei)
        d[fn][st] = min(d[fn][st], wei)
```

```
    for k in range(n):
          for i in range(n):
                for j in range(n):
                      if d[i][k] < INF and d[k][j] < INF:
                               d[i][j] = min(d[i][j], d[i][k] +
d[k][j])

    ans = 0
    for i in range(n):
        for j in range(n):
            ans += d[i][j]
    return str(round(ans/(n*(n-1)) , 2))
```

1.11 Вторая попытка
Задачи по информатике (10 - 11 класс)

Задача 1.11.1 (1 балл)

Условие:

Часто, обсуждая космос, в особенности астрономию, необходимо понимать что такое орбита и как тела по ней движутся. В частности длина орбиты планет солнечной системы позволяет определить длину года. От вас требуется определить длину орбиты с погрешностью до 0.7%. Для решения задачи, считать π≈3.14. Ответ округлить до двух знаков после запятой.

Формат входных данных:

В этой задаче вам заданы два целочисленных параметра эллипса: 1≤a≤ $1 \le a \le 10^9$ — длина большой полуоси, $1 \le b \le 10^9$ — длина малой полуоси. Числа заданы в единственной строке и разделенных пробелом.

Формат выходных данных:

Выведите на экран единственное число -- длину орбиты с точностью до 2 знаков после запятой.

Решение:

Орбита имеет форму эллипса о чем неявно сказано в условии. Для решения задачи воспользуемся формулой длины эллипса, изучаемой в школе. Важно понимать, что формула не является точной.

$$4\pi \frac{ab + (a-b)^2}{a+b}$$

```
a, b =  map(int, input().split())
l = 4.0 *(3.14 * a * b + (a - b)**2)/(a + b)
print(round(l, 2))
```

Задача 1.11.2 (2 балла)

Условие:

Космические корабли обшиты специальным материалом, в состав которого кроме всего прочего входят алюминий и титан. Казалось бы обшивка, состоящая из таких материалов, должна быть надежной. Но надежна она только до тех пор пока она целая. По этой причине при продолжительном использовании аппарата его обшивку необходимо регулярно проверять. Эту задачу может взять на себя специальный манипулятор. Рассмотрим упрощенную модель: Каждый лист обшивки представляет собой параллелограмм на плоскости космического корабля.Манипулятор проверяет за раз несколько точек. В данной задаче манипулятору заданы точки на разных углах параллелограмма(не обязательно все). Точки, которые могут быть заданы отмечены на рисунке зеленым.Кроме тех точек, которые манипуляторы заданы жестко, необходимо проверить дополнительно точку, которая находится в центре симметрии листа обшивки. Напишите программу, которая по входным данным определит координаты центра симметрии листа.

Формат входных данных:

В первой строке задано целое неотрицательное число k≤4 — количество известных пар координат углов листа обшивки. В следующих k строках находится по 2 целых числа x, y разделенных пробелом. −10^9 ≤ x,y ≤ 10^9 — координата очередного угла по оси x и по оси y соответственно.

Формат выходных данных:

В единственной строке выведите:Либо строку "Impossible" для случаев, когда однозначно определить центр симметрии листа невозможно.Либо два числа разделенные пробелом -- координаты x, y центра листа с точностью до 2 знаков после запятой.

Решение:

Для числа углов меньшего, чем 3 решение всегда будет неоднозначное построение исходного ромба.

Для 3 углов будет неоднозначность в случае если данные три угла образуют равносторонний треугольник. Так как невозможно понять какие из ребер являются сторонами ромба, а какое ребро является диагональю. В остальных случаях центр ромба — пересечение его диагоналей.

Остается построить точку пересечения диагоналей. В случае когда выбраны не диагонали, получим параллельные прямые.

Уравнение прямой, проходящей через 2 точки (x1, y1), (x2, y2) Имеет следующий вид

$(y_1 - y_2)\,x + (x_2 - x_1)y + (x_1y_2 - x_2y_1) = 0$

Точка пересечения двух прямых, заданных выражениям $A_1x + B_1y + C_1 = 0$ и $A_2x + B_2y + C_2 = 0$ вычисляется через решение системы из этих двух уравнений.

Если $A_1B_2 = A_2B_1$, то прямые параллельны и необходимо выбрать другую пару вершин.

В ином случае координаты центра можно вычислить по формулам:

$x = (B_2\,C_1 - B_1C_2)\,/\,(A_2B_1 - A_1B_2)$

$y = (A_1C_2 - A_2C_1)\,/\,(A_2B_1 - A_1B_2)$

```
def solve():
    k = int(input())
    if k < 3:
        return "Impossible"
    x1, y1 = map(int, input().split())
    x2, y2 = map(int, input().split())
    x3, y3 = map(int, input().split())
    if k == 3:
     if ((x1 - x2)**2 + (y1 - y2)**2)**0.5 == ((x2 - x3)**2 + (y2 -
y3)**2)**0.5 and ((x1 - x2)**2 + (y1 - y2)**2)**0.5 == ((x3 - x1)**2
+ (y3 - y1)**2)**0.5:
        return "Impossible"

    x4, y4 = map(int, input().split())
    a2, b2 = y2 - y1, x1 - x2
```

```
    c2 = x2 * y1 - x1 * y2
    a1, b1 = y4 - y3, x3 - x4
    c1 = x4 * y3 - x3 * y4
    d = a2 * b1 - a1 * b2;
    if d == 0:
        x2, y2, x3, y3 = x3, y3, x2, y2
        a2, b2 = y2 - y1, x1 - x2
        c2 = x2 * y1 - x1 * y2
        a1, b1 = y4 - y3, x3 - x4
        c1 = x4 * y3 - x3 * y4
        d = a2 * b1 - a1 * b2;

    x = round((b2 * c1 - b1 * c2)/d, 2)
    y = round((a1 * c2 - a2 * c1)/d, 2)
    return "{} {}".format(x, y)
```

Задача 1.11.3 (3 балла)

Условие:

В результате измерений ученые собирают достаточно большое количество данных. Их можно представлять по разному. Однако два наиболее часто используемых варианта: текстовое представление и графическое представление. В этом задании вашей задачей будет перевод информации из текстовой в графическую. В каждом тесте вам будет дан набор данных, по которым потребуется построить вертикальную гистограмму, которая растет вверх.

Формат входных данных:

В первой строке дано целое число N — количество столбцов в гистограмме $0<N\leq100$. В следующих N строках будет дано по паре название -- значение. Название это строка, которая не содержит пробелов. Значение — целое число $0<x\leq300$. Значение показывает высоту столбца гистограммы. Название и значение разделены одним пробелом.

Формат выходных данных:

Выведите гистограмму по входным данным. Порядок столбцов гистограммы должен совпадать с порядком в котором идут столбцы. В случае если на данной высоте в данной колонке гистограмма ещё не закончилась выведите "|". Если закончилась, выведите ".". Как минимум один столбец должен оканчивается на "|".Замечание: Внимательно изучите примеры входных данных и результатов, которые должны быть получены. Если результат вашей программы на тестах из условия не похож на результат теста в примере, ваша программа гарантировано не пройдет тестирование.

Решение:

Задача подразумевает реализацию кода без каких-либо сложных алгоритмов или сложных формул.

В процессе считывания необходимо узнать максимальную высоту столбца. Когда все данные считаны в массив, необходимо для каждого ряда диаграммы вывести корректный символ для каждого столбца. Для того чтоб понять будет ли на i-ой строке столбца присутствовать j-й столбец необходимо сравнить i с j-им элементом массива, который мы визуализируем.

```
def solve():
    n = int(input())
    tmp = dict()
    pos = list()
```

```
    max_col = 0
    res = list()
    for _ in range(n):
        key, value = input().split()
        value = int(value)
        pos.append(value)
        max_col = max(max_col, value)

    for i in range(max_col,0, -1):
        for counter in pos:
            if counter >= i:
                res.append('|')
            else:
                res.append('.')
        res.append('\n')
    return ''.join(res)
```

Задача 1.11.4 (4 балла)

Условие:

Вадим и Саша друзья по переписке. Иногда им очень хочется встретиться и обсудить любимые книги вживую, но расстояние является большой проблемой для школьников. Саша любит говорить, что если можно было выбрать города, которые находятся дальше всего друг от друга, то они выбрали. Ваша задача проверить так ли это. На карте отмечены N городов и M дорог между ними. Для каждой дороги между городами известна ее длина. Ваша задачи узнать наибольшее расстояние между двумя городами такое, что добраться из первого выбранного города во второй невозможно по более короткой дороге.

Формат входных данных:

В первой строке заданы два натуральных числа N≤200,M≤10000. В следующих M строках заданы тройки чисел — два номера городов, которые соединены данной дорогой и длина данной дороги. Города пронумерованы числами от 1 до N. Длины дорог — целые числа, не превышающие 1000. Между парой городов может быть несколько дорог. Гарантируется, что из любого города на карте можно попасть в любой другой, однако не гарантируется, что есть прямую дорога между любой парой городов.

Формат выходных данных:

В единственной строке выведите суммарную длину дороги между парой городов такую, что между этой парой городов нет маршрута с меньшей суммарной длиной, а между любой другой парой городов суммарная длина кратчайшего пути меньше.

Решение:

Дороги и города можно представить в виде графа. Тогда для решения задачи нам необходимо знать расстояние между каждой пары дорог. Один из самых эффективных алгоритмов для решения данной задачи -- алгоритм Флойда Уоршелла, подразумевающий что на k-ом шаге мы знаем правильные расстояния между всеми парами городов для графа, в котором остались только вершины с номерами от 0 до k-1.

На k-ом шаге мы пересчитываем расстояние между каждой парой вершин по формуле: `d[i][j] = min(d[i][j], d[i][k] + d[k][j])`
Фактически мы проверяем не появился ли более выгодный путь между вершинами i и j после добавления в граф вершины номер k и всех ее ребер к уже включенным вершинам.

```
def solve():
    INF = 10**12
```

```
    n, m = map(int , input().split())
    d = list()
    for sm in range(n):
        d.append([INF]*n)
    for tm in range(n):
        d[tm][tm] = 0
    for _ in range(m):
        st, fn, wei = map(int, input().split())
        st -= 1
        fn -= 1
        d[st][fn] = min(d[st][fn], wei)
        d[fn][st] = min(d[fn][st], wei)
    for k in range(n):
         for i in range(n):
              for j in range(n):
                   if d[i][k] < INF and d[k][j] < INF:
                           d[i][j]  =  min(d[i][j],  d[i][k]  +
d[k][j])
    ans = 0
    for i in range(n):
        for j in range(n):
            ans = max(d[i][j], ans)
return ans
```

1.12 Критерии оценки и проходной балл

Оценка решения задач участниками проводилась системой Stepik с учетом количества попыток, затрачиваемых на решение задачи. Расчет балла, начисляющегося за задание, определялся по формуле:

$$\text{Количество баллов за задачу} = \frac{\text{Максимальное количество баллов за задачу}}{\text{Количество попыток}}$$

Из двух попыток по каждому предмету учитывался лучший результат. Результаты по трем предметам суммировались и определяли общий балл по заданиям первого этапа. Проходной балл по первому этапу: 10 баллов.