Профиль «Космические системы» 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, задачи по математике и информатике общие для всех участников. Решение задач по информатике предполагало написание программ, допускалось использовать один из двух языков программирования: Python или C++. На решение задач первого отборочного этапа участникам давалось 3 недели. 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 (математика, физика и информатика) — суммарно от 0 до 30 баллов.

1.1. Задачи по математике

Задача 1.1.1 (1 балл)

В поисках внеземной жизни ученые обнаружили интересный живой организм - Камкохоб. Эксперименты в земных условиях показали, что Камкохоб размножается делением. То есть родительский организм исчезает, и образуются новые особи. При этом каждая особь либо делится ровно на 5 потомков, либо не размножается, а остается одной особью . Экспериментальный образец, привезенный на Землю размножался, некоторые его потомки тоже размножались. **Выберите в таблице**, какое количество потомков могло получиться в итоге, а какое не могло.

Количество потомков	**Могло получиться**	**Не могло получиться**
9		
15		
1001		
2016		

РЕШЕНИЕ:

Правильный ответ на поставленную задачу представлен в таблице. Подробнее ход решения рассматривается в следующей задаче.

Количество потомков	**Могло получиться**	**Не могло получиться**
9	да	
15		да
1001	да	
2016		да

Задача 1.1.2 (1 балл)

В поисках внеземной жизни ученые обнаружили интересный живой организм - Камкохоб. Эксперименты в земных условиях показали, что Камкохоб размножается делением. То есть родительский организм исчезает, и образуются новые особи. При этом каждая особь либо делится ровно на 5 потомков, либо не размножается, а остается одной особью. Экспериментальный образец, привезенный на Землю, размножался, некоторые его потомки тоже размножались. Какое количество потомков могло получиться? **Опишите всю серию возможных ответов общей формулой**, используя переменную x (где x - натуральное число).

РЕШЕНИЕ:

Посмотрим, как меняется общее количество особей при одном размножении. Родительская особь исчезает, появляется 5 новых. То есть общее количество увеличивается на 4. Поскольку исходно у нас одна особь, то могло получиться только число, дающее остаток 1 от деления на 4.

Формула 4x+1, где x - натуральное число, описывает все возможные варианты. Действительно, чтобы получить 4x+1 особей, достаточно размножить любые x особей.

Задача 1.1.3 (3 балла)

Локатор видит часть пространства, ограниченную конусом. Максимальный угол между образующими этого конуса равен 60 градусам. Объекты, находящиеся дальше расстояния a, уже далеко и не попадают в зону видимости локатора. Через ось конуса проведены две перпендикулярные плоскости, которые делят конус на четыре части. В одной из частей обнаружен объект в форме сферы, вписанный в эту часть. Сфера касается обеих перпендикулярных плоскостей, поверхности конуса и его основания. Длина образующей равна a. Найдите радиус этой сферы.

РЕШЕНИЕ:

Пусть P — вершина конуса, O — центр основания, r — радиус сферы с центром Q, касающейся перпендикулярных плоскостей COP и DOP, плоскости основания конуса — в точке *M*, а боковой поверхности конуса ─ в точке *F*. Можно считать, что

OC=OD= $\frac{a}{2}$, OP= $\frac{a\sqrt{3}}{2}$

Через высоту *PO* конуса и параллельную ей прямую *QM* проведём плоскость. Получим осевое сечение конуса ─ равносторонний треугольник *APB* со стороной *a*. Окружность радиуса *r* с центром *Q* вписана в угол *PAO*, поэтому

$AM = QM \text{ ctg } \angle MAQ = r \text{ ctg } 30° = r\sqrt{3}$

Пусть прямая *OQ* образует с плоскостью основания конуса угол α. Тогда

$OM = QM \text{ ctg } \angle MOQ = r \text{ ctg } \alpha$

Поскольку *OA* = *AM* + *OM*, имеем уравнение

$\frac{a}{2} = r\sqrt{3} + r \text{ ctg } \alpha$

Таким образом, для решения задачи достаточно найти ctg α.

Рассмотрим куб с вершиной *L*. Его диагональ, проведённая из вершины *L*, образует с плоскостями трёх граней, содержащих вершину *L*, угол, также равный α. Если ребро куба равно *c*, то его диагональ грани равна $c\sqrt{3}$

Значит, ctg = $c\sqrt{2}/c = \sqrt{2}$

Из уравнения $\frac{a}{2} = r\sqrt{3} + r\sqrt{2}$, находим $r = a\frac{\sqrt{3}-\sqrt{2}}{2}$

Задача 1.1.4 (1 балл)

Незадачливый космонавт Иннокентий почувствовал себя плохо после центрифуги и не может определить направление, он находится в 5 метрах от комиссии и движется по прямой. Каждую секунду он с равной вероятностью либо приближается на метр к ней, либо отдаляется. Если он дошел до комиссии, то больше уже никуда не идет. Найдите вероятность попадания в руки комиссии не позже чем **на 6 секунде**.

РЕШЕНИЕ:

Пусть комиссия находится в пяти метрах справа от космонавта.

Любой путь космонавта за 6 секунд кодируется последовательностью из 6 символов П или Л (П, если он идет направо и Л - если налево). Всего таких последовательностей 64, из них нам годятся две: ПППППП и ПППППЛ.

Значит вероятность попасть к комиссии равна 2/64=1/32

Ответ: 1/32

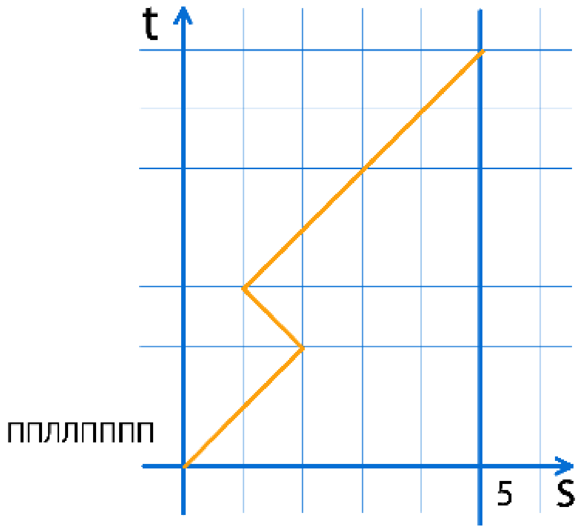
Задача 1.1.5 (2 балла)

Незадачливый космонавт Иннокентий почувствовал себя плохо после центрифуги и не может определить направление, он находится в 5 метрах от комиссии и движется по прямой. Каждую секунду он с равной вероятностью либо приближается на метр к ней, либо отдаляется. Если он дошел до комиссии, то больше уже никуда не идет. Найдите вероятность попадания в руки комиссии не позже чем **на 10-й секунде**.

РЕШЕНИЕ:

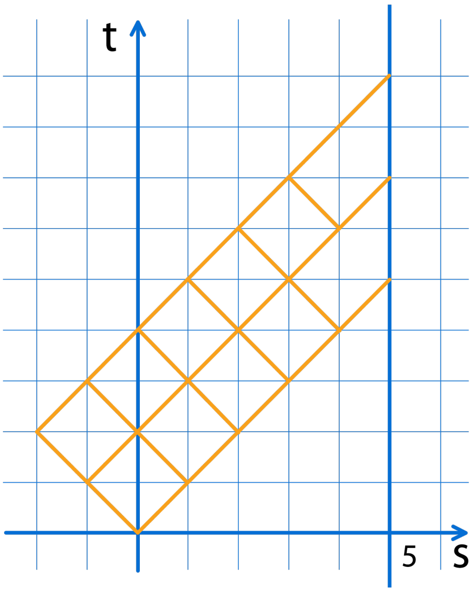
Изобразим путь космонавта на графике. По вертикальной оси отложим время, а по горизонтальной расстояние. Изначально космонавт находится в точке s=0, а комиссия в точке s=5.

На рисунке обозначен путь для последовательности ППЛППППП:

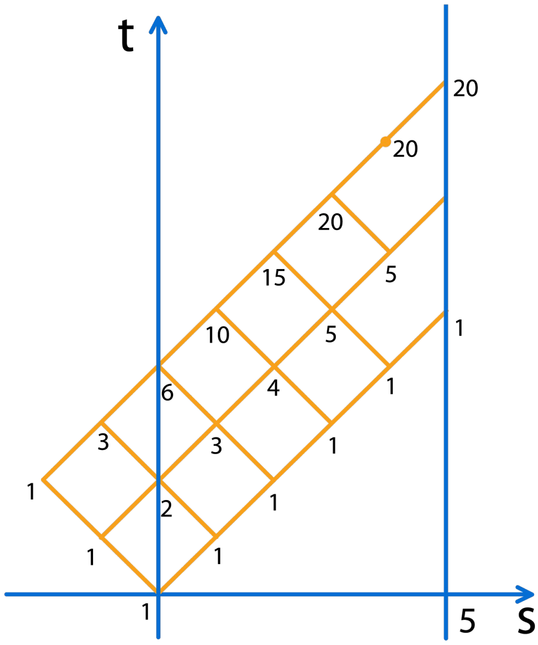


Заметим также, что космонавт может прийти к комиссии только на нечетных шагах. Таким образом, нас интересует, сколько существует путей, ведущих к комиссии за 5, 7 и 9 секунд.

Переформулируем задачу: сколько существует путей, по сетке на рисунке ниже, ведущих к прямой s=5. Ходить можно только двигаясь вверх.



Посчитаем количество таких путей для каждой точки, начиная от начальной.



За 5 шагов приводит 1 путь, за 7 - 5 путей, за 9 - 20 путей.

Значит искомая вероятность равна 1/32+5/128+20/512 = 7/64

Ответ: 7/64

1.2. Задачи по физике (9 класс)

Задача 1.2.1 (2 балла)

Для того, чтобы вернуть спускаемый аппарат массой 10 кг с поверхности Луны, было предложено использовать пушку, стреляющую аппаратом вертикально вверх. Оцените минимальное количество взрывчатки для запуска аппарата с поверхности Луны на высоту $H = R$ над поверхностью Луны, где R - радиус Луны (1735,97 км). Считайте теплоту сгорания взрывчатки равной 1 кКал/г. Ускорение свободного падения на поверхности Луны считайте равным g=1.62м/с^2. Запишите массу взрывчатки, необходимую для запуска аппарата на высоту H. Ответ дайте в килограммах, с точностью до десятых.

РЕШЕНИЕ:

Находим энергию гравитационного поля.

$$E = \frac{-GMm}{r}$$

Находим энергию необходимую для подъёма.

$$E_{треб} = \frac{-GMm}{2R} + \frac{GMm}{R} = \frac{GMm}{2R}$$

Находим необходимое кол-во топлива

$$M_{топлива} = \frac{GMm}{2RQ_{сгорания}} \approx 3{,}3 \text{ кг}$$

Задача 1.2.2 (1 балл)

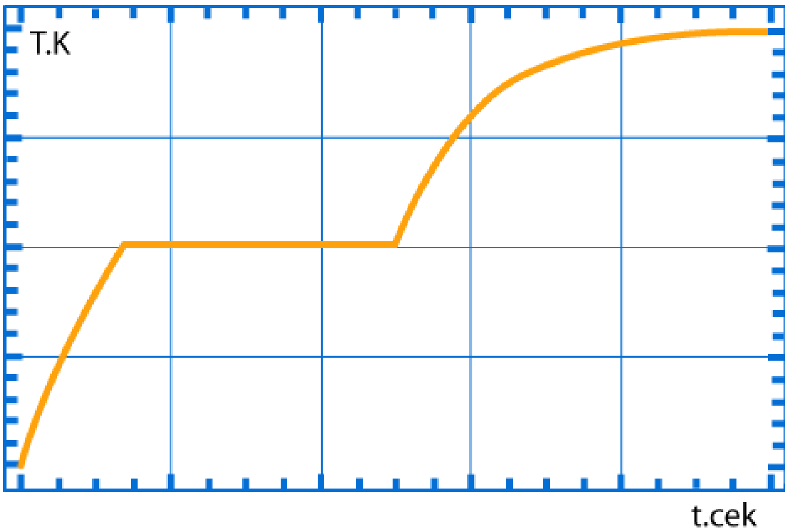
Космический корабль представляет собой цилиндр радиуса *2R*. Вокруг него катается внешний цилиндрический модуль радиуса *R*, каждый день совершая полный «объезд». Сам корабль вращается вокруг планеты с периодом 100 дней. Сколько оборотов зафиксирует счетчик оборотов через 100 дней полёта? Запишите число оборотов модуля за 100 дней полета.

РЕШЕНИЕ:

Совершая оборот вокруг корабля модуль совершает не 2 а 3 оборота, важно посмотреть на то сколько проезжает центр. Значит за сто дней будет пройдено 300 оборотов. Но важно что при этом была совершен ещё один оборот вокруг планеты. Итого 301.

Задача 1.2.3 (2 балла)

В ходе эксперимента по выращиванию кристалла сломалась система охлаждения, и уже выращенный образец расплавился. Для изучения остался только график зависимости температуры от времени, но и он был поврежден. Восстановите цену делений на этом графике. Известно что теплопередача пропорциональна разнице температур, внешняя температура 400 К. Про материал известны: температура плавления 300 К, теплота плавления λ=24кКал/кг, теплоемкость жидкой фазы Сж=8000кал/кг$*$К, коэффициент теплопроводности 32 Дж/$\circ$. Масса образца 1 кг.



Запишите цену делений **шкалы времени**. Ответ дайте в **секундах** с точностью до целых.

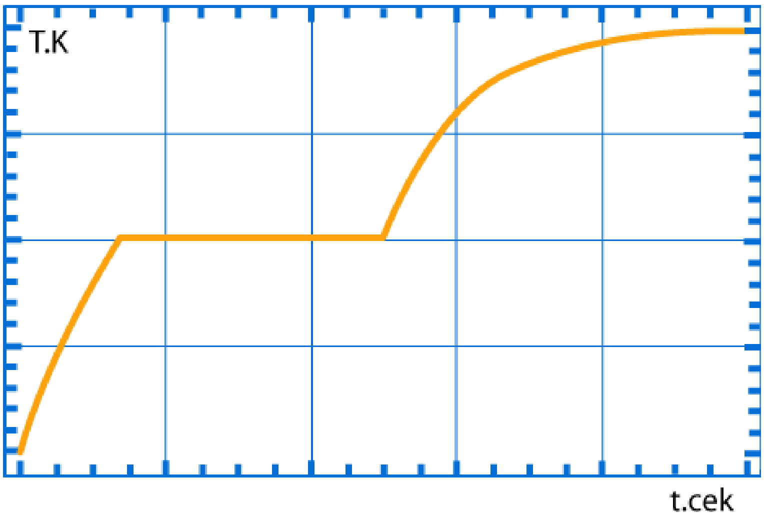
РЕШЕНИЕ:

Из плато на графике и касательной в самой верхней точке легко восстановить шкалу температур. Т.к. плато соответствует плавлению, а в самой верхней точке температура практически вышла на константу, т.е. сравнялась с внешней. Отсюда шаг шкалы температур 10˚С.

А зная шаг температуры и параметры тела легко вычислить и шаг по времени, например через время плавления - 10 с.

Задача 1.2.4 (3 балла)

В ходе эксперимента по выращиванию кристалла сломалась система охлаждения, и уже выращенный образец расплавился. Для изучения остался только график зависимости температуры от времени, но и он был поврежден. Восстановите цену делений на этом графике. Известно что теплопередача пропорциональна разнице температур, внешняя температура 400 К. Про материал известны: температура плавления 300 К, теплота плавления λ=24кКал/кг, теплоемкость жидкой фазы Сж=8000кал/кг∗К, коэффициент теплопроводности 32 Дж/∘. Масса образца 1 кг.



Запишите цену делений **шкалы температуры**. Ответ дайте в **градусах**, с точностью до целых.

РЕШЕНИЕ:

Из плато на графике и касательной в самой верхней точке легко восстановить шкалу температур. Т.к. плато соответствует плавлению, а в самой верхней точке температура практически вышла на константу, т.е. сравнялась с внешней. Отсюда шаг шкалы температур 10˚С.

Задача 1.2.5 (4 балла)

Спутник, находящийся на высоте Н = 7000 км от центра планеты с массой М= 1,076*10^{23} кг, переходит на точно такую же по высоте орбиту, пересекающую изначальную под углом α=5º. Найдите силу, с которой действовали маневровые двигатели спутника, если известно, что поворот занял t=30 с, а масса спутника m=50 кг. Ответ дайте в ньютонах с точностью до целых.

РЕШЕНИЕ:

Сначала найдем скорость спутника:

$V_0 = \sqrt{\frac{GM}{H}}$, теперь найдем скорость поправки к движению (основание равностороннего треугольника.

$$\Delta V = V \sin \alpha$$

Т.к. время маневра крайне мало по сравнению с периодом обращения можно оценить среднюю силу как:

$$F = m\frac{\Delta V}{t} = m\frac{\sqrt{\frac{GM}{H}}\sin\alpha}{t} \approx 147{,}2 \text{ Н}$$

1.3. Задачи по физике (10-11 класс)

Задача 1.3.1 (2 балла)

Атмосфера Венеры агрессивная и очень горячая (температура на поверхности около $T = 750$ К), поэтому, даже при успешной посадке, обеспечить длительную работу исследовательской станции на планете достаточно сложно. Для того, чтобы продлить время работы на поверхности для одной из исследовательских станций, предложили часть полезной мощности передать охлаждающей установке. Какую минимальную часть энергии необходимо передать на охлаждение, если о станции известно следующее: станция представляет собой шар радиуса $R = 0{,}5$ м, теплопроводность ее поверхности $\varkappa = 10$ Вт/К*м^2? Полная вырабатываемая станцией мощность $P = 8$ кВт. Температура, при которой приборы выходят из строя, равна $T_{крит} = 77$°C.

Ответ дайте в виде десятичной дроби, округлив до сотых.

РЕШЕНИЕ:

Найдем поступление тепла через стенки:

$$\Phi = k * (T_{гор} - T_{хол}) * S$$

Это поступление необходимо компенсировать с помощью холодильной установки. Мощность холодильной установки можно оценить с помощью цикла Карно. Найдем холодильный коэффициент:

$$\eta_{холодильника} = \frac{T_{хол}}{T_{гор} - T_{хол}}$$

Тогда: $\eta_{холодильника} P_{хол} = \Phi$, откуда можно получить искомую величину:

$$\frac{P_{хол}}{P} \approx 0{,}574$$

Задача 1.3.2 (2 балла)

Экспедиция на Марс пытается найти воду в подземных полостях, используя метод, связанный с измерением периода колебаний маятника и его изменением вблизи неоднородностей плотности планеты. Встроенная в марсоход система, работающая подобным образом, настроена на базовую частоту $\omega_0 = 5{,}7966$ рад/с, умеет измерять период маятника с точностью $\Delta t = 1{,}000 * 10^{-4}$ с. Оцените минимальный радиус сферической полости, которая может быть обнаружена. Считайте, что Марс представляет собой идеальный равномерно плотный шар, за исключением сферической, заполненной водой полости, находящейся непосредственно под поверхностью планеты. Ускорение свободного падения на поверхности Марса можно считать равным 3,36 $м/c^2$, средняя плотность Марса 3,933 г/см³, плотность воды 1,000 г/см³, гравитационная постоянная G= 6,674* 10^{-11} Н·м²/(кг²). Ответ дать в метрах с точностью до целых.

РЕШЕНИЕ:

Зная базовую частоту можно найти характерную длину маятника, который установлен в измерительной системе

$$\omega_0 = \sqrt{\frac{g}{l}} \quad => \quad l = \frac{g}{\omega_0^2}$$

Для того чтоб ы понять, какую поправку вносит полая сфера, воспользуемся приёмом отрицательной массы. За счет «отрицательной ми ассы» снизится g и соответственно снизится и период, найдем минимальное снижение g, которое можно обнаружить

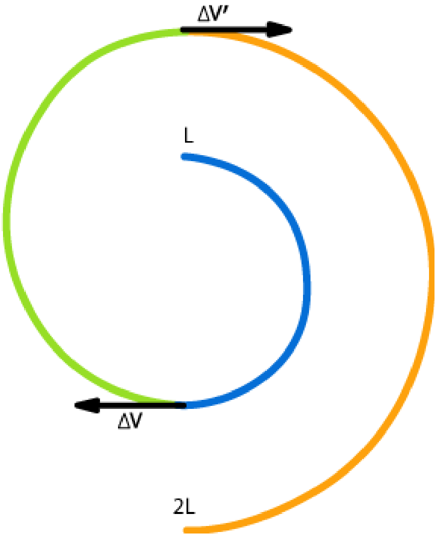
$$\Delta g = T_0^2 l - (T_0 - \tau)^2 l$$

Рассчитаем радиус из закона всемирного тяготения. Эту поправку внесёт сфера радиуса

$$R = \frac{\Delta g * 3}{4\pi * \rho * G} \approx 125 \text{м}$$

Задача 1.3.3 (2 балла)

Спутник массы m=100 кг, находящийся на орбите радиуса L=2500 км, обращаясь вокруг планеты массы M= 4,8017· 10^{22} кг , разгоняясь, совершает эллиптический переход на орбиту радиуса 2L=5000 км. **Найдите время, требующееся на переход.** Изменения скорости, производимые в апоцентре и перицентре, считать мгновенными. Ответ дайте в часах, с точностью до десятых.



РЕШЕНИЕ:

Используя закон всемирного тяготения найдем скорости спутника на орбитах, считая орбиты окружностями.

$$V_1 = \sqrt{\frac{GM}{L}} \text{ и } V_2 = \sqrt{\frac{GM}{2L}}$$

Из рисунка видно, что во время маневра спутник движется по эллипсу с большой полуосью 3/2L. С помощью третьего закона Кеплера можно найти период обращения по такому эллипсу:

$T = \left(\frac{1{,}5L}{L}\right)^{3/2} T_0$, где $T_0 = \frac{2\pi L}{V_1}$ - период обращения вокруг планеты по орбите радиуса L.

Искомое время равно половине периода T = 3.54ч

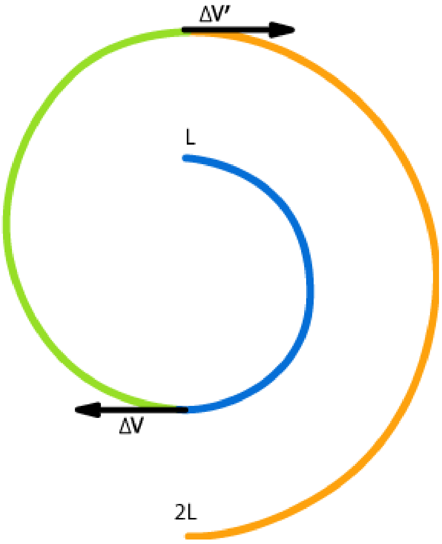
Из изменения полной механической энергии найдем работу двигателей:

$$\frac{V_0^{\ 2}}{2} - \frac{GM}{R} = \frac{V_1^2}{2} - \frac{GM}{R_1} + A$$

A = -32150Дж

Задача 1.3.4 (3 балла)

Спутник массы m=100 кг, находящийся на орбите радиуса L=2500 км, обращаясь вокруг планеты массы M= $4{,}8017 \cdot 10^{22}$ кг, разгоняясь, совершает эллиптический переход на орбиту радиуса 2L=5000 км. **Найдите работу, совершенную двигателями.** Изменения скорости, производимые в апоцентре и перицентре, считать мгновенными.



Ответ дайте килоджоулях с точностью до целых.

РЕШЕНИЕ:

Используя закон всемирного тяготения найдем скорости спутника на орбитах, считая орбиты окружностями.

$V_1 = \sqrt{\dfrac{GM}{L}}$ и $V_2 = \sqrt{\dfrac{GM}{2L}}$

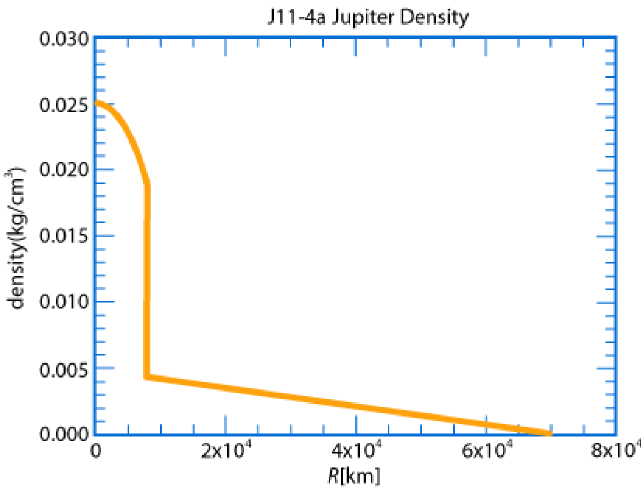
Из изменения полной механической энергии найдем работу двигателей:

$$\frac{V_0^{\,2}}{2} - \frac{GM}{R} = \frac{V_1^2}{2} - \frac{GM}{R_1} + A$$

A = -32150Дж

Задача 1.3.5 (3 балла)

Ниже приведен график зависимости плотности атмосферы Юпитера от высоты. Считая ускорение свободного падения g постоянным g=25м/ c^2, найдите глубину погружения, если зонд спускается с высоты H1=7∗ 10^4 км вниз без начальной скорости. Среднюю плотность зонда считайте равной ρ=1.49г/ $см^3$.



Ответ дайте в километрах и округлите до тысяч.

РЕШЕНИЕ:

Находим условие максимального погружения. Вся энергия перешла в потенциальную, кинетическая энергия 0. При погружении она с одной стороны увеличивается (за счёт гравитации), а с другой уменьшается — за счёт погружения в плотные слои атмосферы.

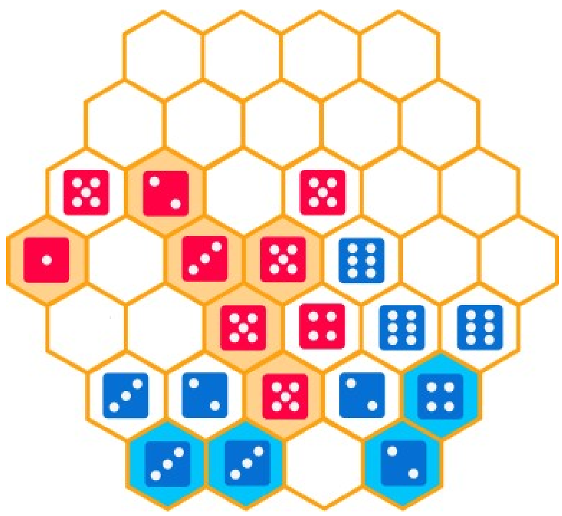
$$\frac{mV^2}{2} = A_{apx} - mg(h - h_1)$$

Первый член в правой части придётся искать методом численного интегрирования. Из графика находим первый, второй просто рассчитывается из начальных данных.

Глубина погружения 31000 км.

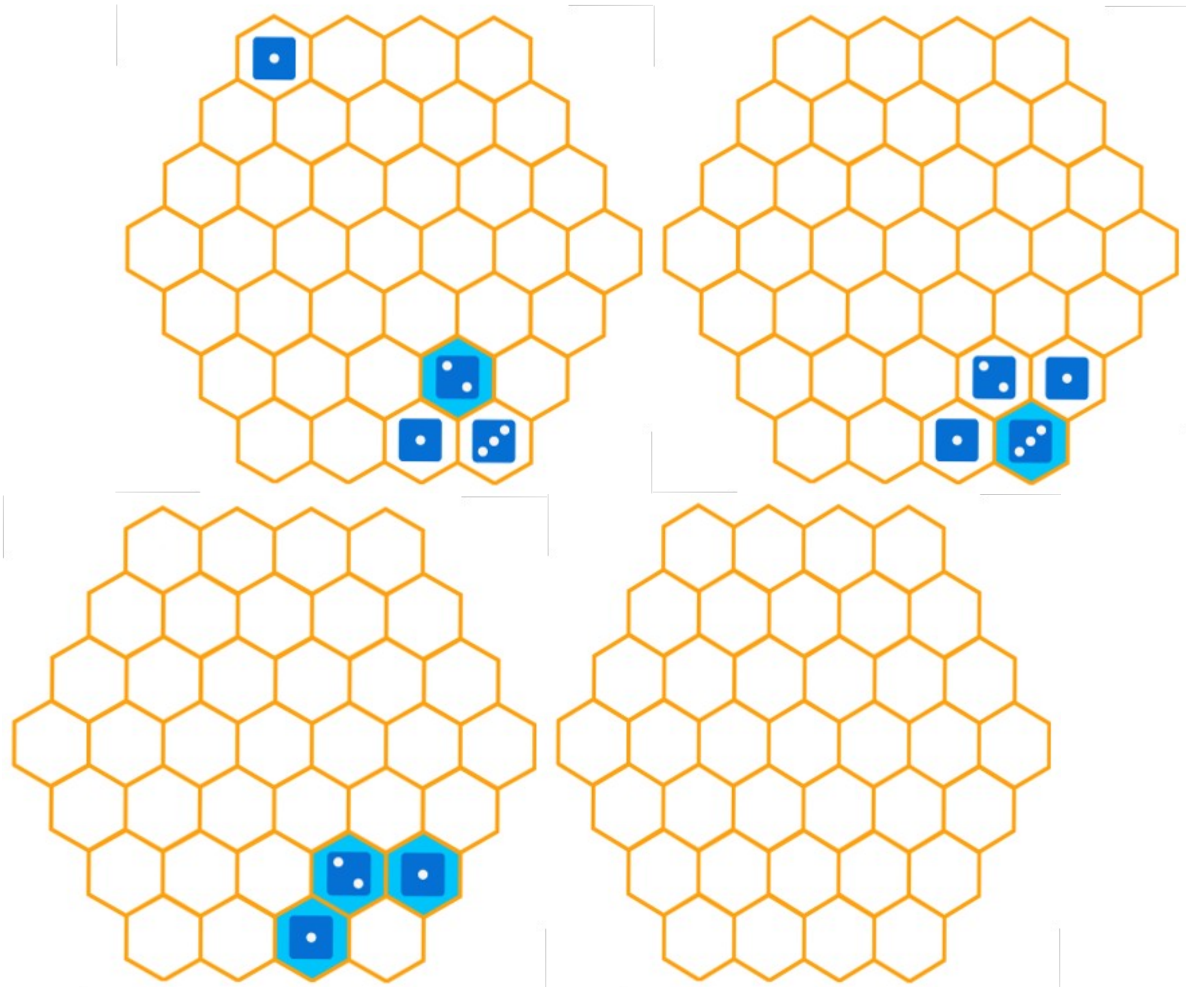
1.4. Задачи по информатике

Задача 1.4.1 «Сибирские кости» (1 балл)



Сибирские кости — абстрактная симметричная настольная игра для двух игроков. Игра ведется на гексагональном поле со стороной четыре клетки обычными игральными костями двух цветов. Кости, у которых количество соседей равно числу на верхней грани, называются окруженными. Цель обоих игроков — получить шесть окруженных костей своего цвета. Выше изображена позиция, в которой выиграл красный игрок, окружённые кости выделены фоном.

Но мы будем рассматривать упрощенную версию этой игры, а именно вариант для одного игрока. В нем необходимо переместить одну кость так, чтобы убрать с поля наибольшее число костей. После хода одновременно убираются все окруженные кости. Если после этого появились новые окруженные кости, то они так же убираются. Этот процесс длится до тех пор, пока на поле остаются окруженные кости. Ниже изображен итог перемещения единицы из левого верхнего угла к группе остальных кубиков.



Формат входных данных:

В каждой строке входных данных задаётся одна строка поля. В первой строке даётся четыре разделённых пробелом неотрицательных числа, не превышающих 6 — начальное состояние первой строки поля. Ноль означает отсутствие кубика, ненулевое число — значение на кубике. Во второй строке даётся пять чисел — состояние второй строки в аналогичном формате. В последующих строках описываются оставшиеся часть поля.

Пример ввода:

```
1 0 0 0
0 0 0 0 0
0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 2 0
0 0 1 3
```

Формат выходных данных:

Нужно вывести единственное неотрицательное целое число - количество кубиков, которое можно снять с поля.

Пример вывода:

```
4
```

РЕШЕНИЕ:

Для решения задачи достаточно аккуратно реализовать написанный в условии задачи алгоритм. Пример программы, реализующей данный алгоритм на языке Python:

```
import copy
import sys

def neighbours(a, x, y):
    res = 0
    for dx, dy in [(-1, -1),(-1, 0),(0, 1),(1, 1),(1, 0),(0, -1)]:
        xx = x + dx
        yy = y + dy
        res +=  int(-1<xx<len(a) and -1<yy<len(a[xx]) and a[xx][yy]>0)
    return res

def clear(a):
    b = []
    res = 0
    for x in range(len(a)):
        b.append([-1] * len(a[x]))
        for y in range(len(a[x])):
            if a[x][y] == 0:
                b[x][y] = 0
                continue
            if a[x][y] != neighbours(a, x, y):
                b[x][y] = a[x][y]
            else:
                b[x][y] = 0
                res += 1
    return (res, b)

```

```
def test(a, x1, y1, x2, y2):
    a = copy.deepcopy(a)
    a[x2][y2] = a[x1][y1]
    a[x1][y1] = 0
    res = 0
    while True:
        q, a = clear(a)
        if q == 0:
            return res
        res += q

def split(s):
    return list(map(int, s.strip().split()))

def solve(dataset):
    lines = dataset.strip().split('\n')

    a = []
    a.append(split(lines[0]))
    a.append(split(lines[1]))
    a.append(split(lines[2]))
    a.append(split(lines[3]))
    a.append([-1] + split(lines[4]))
    a.append([-1, -1] + split(lines[5]))
    a.append([-1, -1, -1] + split(lines[6]))

    ans = 0
    for x in range(len(a)):
        for y in range(len(a[x])):
            if a[x][y] <= 0:
                continue
            for xx in range(len(a)):
                for yy in range(len(a[xx])):
                    if a[xx][yy] != 0:
                        continue
                    t = test(a, x, y, xx, yy)
                    if t > ans:
                        ans = t
    return str(ans)

solve(sys.stdin.read())
```

Задача 1.4.2 «Ломаная» (3 балла)

На доске была нарисована замкнутая ломаная, причём каждая пара соседних звеньев образовывала прямой угол, и все звенья были параллельны одной из сторон доски. Любая пара звеньев имела не более одной общей точки, а так же никакие две вершины не совпадали. Координаты вершин ломаной записали в случайном порядке, а ломаную стёрли. Ваша задача — восстановить саму ломаную.

Формат входных данных:

В первой строке дано чётное натуральное число N — количество вершин ломаной ($4 \leq N \leq 100000$). В последующих N строках даны пары целых чисел, разделённых ровно одним пробелом — координаты вершин. Все координаты не превышают 1000000 по абсолютному значению.

Пример ввода:

```
4
1 0
1 1
0 0
0 1
```

Формат выходных данных:

Выведите номера вершин в том порядке, в котором их нужно соединить, чтобы получить первоначальную ломаную. Нумерация вершин с нуля, в том порядке, в котором они даны во входных данных. Если возможно несколько ответов, выведите любой.

Пример вывода:

```
0 2 3 1
```

РЕШЕНИЕ:

Можно показать, что на каждой вертикальной и горизонтальной прямой будет находиться чётное число точек из входных данных. Понятно, что нижняя (левая) точка обязана соединяться со следующей за ней.

Пример программы, реализующей данный алгоритм на языке Python:

```
import ast
import sys

class Point:
    def __init__(self, x = 0, y = 0):
        self.x = x
        self.y = y
    def turn(self, p):
        return Point(self.smul(p), self.smul(p.norm()))
    def smul(self, p):
        return self.x * p.x + self.y * p.y
    def norm(self):
        return Point(self.y, -self.x)
    def __lt__(self, p):
        if self.x != p.x:
            return self.x < p.x
        return self.y <= p.y
    def area(self, p):
        return (self.x - p.x) * (self.y + p.y)
    def vec(self, p):
        return Point(self.x - p.x, self.y - p.y)

def solveForDirection(points):
    n = len(points)
    perm = [i for i in range(n)]
    perm.sort(key=lambda k: points[k])
    neighbours = [None] * n
    for i in range(1, n):
        if points[perm[i]].x == points[perm[i - 1]].x and neighbours[perm[i
```

```
- 1]] is None:
            neighbours[perm[i - 1]] = perm[i]
            neighbours[perm[i]] = perm[i - 1]
    for p in neighbours:
        if p is None:
            return None
    return neighbours

def getNeighbours(points):
    neighbours = [[], []]
    neighbours[0] = solveForDirection(points)
    points = [p.norm() for p in points]
    neighbours[1] = solveForDirection(points)
    if neighbours[0] is None or neighbours[1] is None:
        return None
    return list(zip(neighbours[0], neighbours[1]))

def getPolygon(neighbours, first):
    ans = [first, neighbours[first][1]]
    while ans[-1] != first:
        x = ans[-2]
        y = ans[-1]
        if neighbours[y][0] == x:
            ans.append(neighbours[y][1])
        else:
            ans.append(neighbours[y][0])
    return ans[:-1]

def solveForLine(points):
    neighbours = getNeighbours(points)
    if neighbours is None:
        return None
    return getPolygon(neighbours, 0)

def solve(dataset):
    lines = dataset.strip().split('\n')
    n = int(lines[0])
    lines = lines[1:]
    points = []
    for line in lines:
        x, y = list(map(int, line.strip().split()))
        points.append(Point(x, y))
    perm = solveForLine(points)
    return ' '.join(map(str, perm))

solve(sys.stdin.read())
```

Задача 1.4.3 «Полет» (6 баллов)

На высоте h метров над поверхностью планеты находится тело. В начальный момент времени его скорость равна $\vec{v}$ метров в секунду. Вычислите время до момента касания телом поверхности планеты, если известно, что:

1. h в начальный момент времени не превышает десяти километров, а мгновенная скорость — одного километра в секунду.
2. На тело действует сила притяжения планеты. Ускорение, создаваемое этой силой,

равно $\frac{GM}{R^2}$, и направлено к центру планеты. Здесь G — гравитационная постоянная, M — масса планеты, а R — расстояние от центра планеты до тела. Величину GM считайте равной 4×10^{13} , а радиус планеты — 6300000 метрам.

3. Так же на тело действует сила аэродинамического сопротивления. Можете считать, что ускорение, создаваемое этой силой, равно $-1.05\rho|\vec{v}|\vec{v}$, где ρ — плотность атмосферы на заданной высоте, выражаемая формулой $\rho(h)= e^{\frac{-h}{9300}}$., $\vec{v}$ — мгновенная скорость тела, а $|\vec{v}|$ — модуль мгновенной скорости.
4. При данных ограничениях смещение вдоль касательной к поверхности планеты много меньше её радиуса планеты, поэтому кривизной планеты следует пренебречь.

Мы понимаем, что аналитически решить эту задачу сложно, поэтому примем ответ с абсолютной погрешностью 0.1секунды. И предупреждаем, что слишком сильно менять состояние тела — не очень хорошая идея.

Формат входных данных:

В первой строке дано одно натуральное число h — высота тела в начальный момент времени, выраженная в метрах. Во второй строке даны два целых числа (v_x , v_y) — координаты начальной скорости тела в метрах в секунду. Ось Oy при этом направлена в сторону, противоположную направлению силы гравитации, а ось Ox — перпендикулярна оси Oy.

Пример ввода 1:

```
100
10 10
```

Пример ввода 2:

```
1
0 0
```

Формат выходных данных:

Выведите единственное вещественное число — время до касания телом поверхности планеты, выраженное в секундах.

Пример вывода1 :

```
105.3283038851984
```

Пример вывода 2:

```
1.6649107662192795
```

РЕШЕНИЕ:

В этой задаче нужно было реализовать любой численный метод, который не вносит слишком большой ошибки. Например, моделирование с изменяемой шириной шага по времени.

```
import math
import sys

GM = 4e13
R = 6300000

def solve(dataset):
    lines = dataset.split('\n')

    h = float(lines[0])
    vx, vy = list(map(float, lines[1].split()))

    x = 0
    t = 0
    dT = 0.1
    while h > 0:
        v = math.sqrt(vx * vx + vy * vy)
        g = -GM / (R + h) ** 2
        p = math.exp(-h/9300) * 1.05 * v
        ax = -p * vx
        ay = -p * vy + g

        dt = dT
        a = math.sqrt(ax * ax + ay * ay)
        if a > 0.01:
            dt = min(dt, 0.01 / a)
        if v > 0.01:
            dt = min(dt, 0.01 / v)

        vx += ax * dt
        vy += ay * dt

        x += vx * dt
        h += vy * dt

        t += dt
    return str(t)
solve(sys.stdin.read())
```

1.5. Критерии определения призеров и победителей

Количество баллов, набранных при решение всех задач суммируется. Призерам первого отборочного этапа было необходимо набрать 10 баллов. Победители первого отборочного этапа должны были набрать 26 баллов (для участников из 9 класса) и 28 баллов (для участников из 10-11 классов).