



*Эффективная и слаженная
командная работа – залог успеха!*

XVII РЕСПУБЛИКАНСКАЯ КОМАНДНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ ПО ПРОГРАММИРОВАНИЮ

с. Чапаево
25–29 февраля 2020 г.

XVII Республиканская командная олимпиада школьников по программированию. — с. Чапаево, 2020.

Сборник содержит условия задач XVII Республиканской командной олимпиады школьников по программированию и возможные варианты решений. Олимпиада проводилась 25–29 февраля 2020 г. на базе Малой академии наук РС (Я) в с. Чапаево под Якутском. Участникам было предложено за пять часов решить девять задач.

СПОНСОРЫ ОЛИМПИАДЫ



ООО «МАЙТОНА»
Генеральный директор —
Александра Сергеевна БАИШЕВА



ООО «АКСИОМА»
Директор —
Наталья Леонтьевна МАХОНИНА

ОРГКОМИТЕТ ОЛИМПИАДЫ

ФЕДОРОВ Михаил Прокопьевич

*проректор по педагогическому образованию
Северо-Восточного федерального университета
имени М. К. Аммосова» — председатель*

ПАВЛОВ Василий Климович

ректор Малой академии наук РС (Я) — заместитель председателя

ТИХОНОВ Владимир Иванович

заместитель министра Министерства образования и науки РС (Я)

НИКОЛАЕВА Наталья Васильевна

*зав. кафедрой теории и методики обучения информатике ИМИ СВФУ,
зав. кафедрой информатики Малой академии наук РС (Я)*

БУДИКИНА Людмила Евсеевна

начальник учебного отдела Малой академии наук РС (Я)

КОРЯКИНА Туйара Васильевна

методист кафедры информатики Малой академии наук РС (Я)

ЖЮРИ ОЛИМПИАДЫ

ПАВЛОВ Никифор Никитич

*к. ф.-м. н., доцент кафедры информационных технологий
ИМИ СВФУ – председатель*

ЛЕВЕРЬЕВ Владимир Семенович

старший преподаватель кафедры ИТ ИМИ СВФУ

НИКОЛАЕВА Наталья Васильевна

*к. ф.-м. н., зав. кафедрой ИТ ИМИ СВФУ,
зав. кафедрой информатики МАН РС(Я)*

ПАВЛОВ Александр Викторович

к. ф.-м. н., доцент кафедры ИТ ИМИ СВФУ

ХОВРОВ Дмитрий Евгеньевич

студент ИМИ СВФУ

ЭВЕРСТОВ Владимир Васильевич

старший преподаватель кафедры ИТ ИМИ СВФУ

КОРЯКИНА Туйара Васильевна

методист кафедры информатики Малой академии наук РС (Я), секретарь жюри

СПИСОК УЧАСТНИКОВ

1. Команда Чурапча-1 МБОУ «Чурапчинская гимназия имени С. К. Макарова», руководитель Захаров Прокопий Прокопьевич
Илларионов Артем Иванович, 9 класс
Иванов Радомир Антонович, 8 класс
Куличкин Тимур Федорович, 8 класс
2. Команда «Чурапча-2», МБОУ «Чурапчинская гимназия имени С. К. Макарова», руководитель Захаров Прокопий Прокопьевич
Чабылысов Ньургун Романович, 8 класс
Корякин Максим Николаевич, 8 класс
Барашков Владислав Владимирович, 9 класс
3. Команда «СПТЛИ-1», МБОУ «Сунтарский политехнический лицей-интернат», руководитель Фролова Светлана
Александр Иванович, 8 класс
Андреев Игорь Константинович, 10 класс
Егоров Айтал Николаевич, 8 класс
4. Команда «СПТЛИ-2», МБОУ «Сунтарский политехнический лицей-интернат», руководитель Фролова Светлана Марковна
Алексеева Анастасия Александровна, 11 класс
Спиров Артемий Александрович, 10 класс
Мекумянов Димитрий Леонидович, 8 класс
5. Команда «СПТЛИ-3» МБОУ «Сунтарский политехнический лицей-интернат», руководитель Гаврильева Лена Ивановна.
Егоров Владислав Евгеньевич, 8 класс
Гаврильев Дамир Гаврильевич, 7 класс
Макаров Мирон Егорович, 7 класс
6. Команда «otl МБНОУ "Октемский НОЦ»,
руководитель Ковров Феликс Филиппович
Корякин Сергей Сергеевич, 11 класс
Терентьев Данил Степанович, 11 класс
Ксенофонтов Василий Ильич, 11 класс
7. Команда «3,1415», МБНОУ «Октемский НОЦ»,
руководитель Ковров Феликс Филиппович.
Егорова Дайаана Андреевна, 9 класс
Васильева Анна Петровна, 9 класс

Васильева Анна Петровна, 9 класс

8. Команда «Покровск-1», МБОУ «Покровская СОШ №3 - ОЦ с УИОП»,
руководитель Соловьева Ирина Васильевна.
Капитонов Айдын Владиславович, 9 класс
Яковлева Арина Вадимовна, 10 класс
Родионов Анатолий Анатольевич, 10 класс
9. Команда «Покровск-2», МБОУ «Покровская СОШ №2 с УИОП»,
руководитель Мордовской Денис Андреевич.
Полторыхина Мария Станиславовна, 8 класс
Николаева Сахааи Владимировна, 7 класс
Гуськов Данил Алексеевич, 7 класс
10. Команда «Нюрба», МБОУ «Нюрбинский технический лицей им. А.Н. Чу-
совского», руководитель Петров Николай Николаевич
Иванов Александр Прокопьевич, 10 класс
Иванов Александр Прокопьевич, 11 класс
Семенов Ефим Николаевич, 8 класс
11. Команда «Эрэл-1», ГБНОУ РС(Я) «Республиканский лицей-интернат»,
руководитель Данилов Михаил Николаевич.
Захаров Алексей Викторович, 11 класс
Зыков Тимур Алексеевич, 10 класс
Семенов Никита Сергеевич, 10 класс
12. Команда «Эрэл-2», ГБНОУ РС(Я) «Республиканский лицей-интернат»,
руководитель Данилов Михаил Николаевич.
Соров Айаал Львович, 10 класс
Жирков Айал Андреевич, 10 класс
Дмитриев Николай Николаевич, 10 класс
13. Команда «Эрэл-3», ГБНОУ РС(Я) «Республиканский лицей-интернат»,
руководитель Данилов Михаил Николаевич.
Кузьмин Андрей Степанович, 10 класс
Павлов Никифор Михайлович, 9 класс
Уваровская Лена Александровна, 9 класс
14. Команда «Эрэл-4», ГБНОУ РС(Я) «Республиканский лицей-интернат»,
руководитель Данилов Михаил Николаевич
Семенов Дархан Рустамович, 7 класс
Ноговицын Айтал Евгеньевич, 7 класс
Васильев Ян Евгеньевич, 7 класс

15. Команда «MaуaLyceum», МБОУ «Майинский лицей им. И.Г. Тимофеева
МР “Мегино-Кангаласский улус”»,
руководитель Харитонова Мария Иннокентьевна
Оконешников Иван Валентинович, 11 класс
Иванов Ян Андреевич, 11 класс
Игнатъев Михаил Александрович, 10 класс
16. Команда «СУНЦ СВФУ 1», СУНЦ СВФУ,
руководитель Алексеев Станислав Капитонович
Ефимова Валерия Афанасьевна, 11 класс
Васильевна, 11 класс
Андреев Игорь Константинович, 10 класс
17. Команда «СУНЦ СВФУ 2», СУНЦ СВФУ,
руководитель Алексеев Станислав Капитонович
Алексеев Станислав Капитонович, 10 класс
Николаева Юлия Кирилловна, 10 класс
Макаров Ньургун Михайлович, 11 класс
18. Команда «Нерюнгри», МОУ «Гимназия №1 г.Нерюнгри имени С. С. Каримовой», руководитель Самсонова Юлия Васильевна.
Сухих Дмитрий Сергеевич, 10 класс
Амвросов Артем Игоревич, 8 класс
Гуреев Никита Андреевич, 11 класс
19. Команда «ЯКУТСК-1», МОБУ ФТЛ, руководитель Куличкин Н. Н.
Захаров Айтал, 8 класс
Захаров Айтал, 7 класс
Павловский Владислав, 8 класс
20. Команда «ЯКУТСК-2», МОБУ ФТЛ, руководитель Куличкин Н. Н.
Лаптев Максим Сергеевич, 10 класс
Хуанг Ян Минцзиевич, 10 класс
Николаев Роберт Михайлович, 10 класс
21. Команда «ЯКУТСК-3», МОБУ ЯГЛ,
руководитель Жиркова Марта Маратовна.
Кангельдиева Асель, 9 класс
Поскачин Владимир, 9 класс
Федорова Ева, 9 класс
22. Команда «ЯКУТСК-4», МАОУ Саха политехнический лицей,
руководитель Герасимова Галина Егоровна

- Колодезников Егор, 11 класс*
Туприн Айтал, 9 класс
Петров Тимур, 9 класс
23. Команда «ЯКУТСК-5», МОБУ ФТЛ, руководитель Куличкин Н. Н.
Карпов Вячеслав Ньургуновна, 10 класс
Карпов Вячеслав, 10 класс
Баишев Василий, 10 класс
24. Команда «ЯКУТСК-6», МОБУ ЯГЛ,
 руководитель Жиркова Марта Маратовна
Афонская Анжела, 9 класс
Давыдов Вася, 9 класс
Китаев Богдан, 9 класс
25. Команда «ЯКУТСК-7», МОБУ СОШ31,
 руководитель Олесова Софья Афанасьевна
Олесова Софья Афанасьевна, 10 класс
Никифорова Айталиа Ньургуновна, 9 класс
Винокурова Кристина Сергеевна, 8 класс
26. Команда «ЯКУТСК-8», МАОУ Саха политехнический лицей,
 руководитель Герасимова Галина Егоровна
Кузнецов Юра, 8 класс
Егоров Айтал, 7 класс
Степанов Семен Георгиевич, 9 класс
27. Команда «ЯКУТСК-9» МОБУ ФТЛ, руководитель Романов Юрий Николаевич
Местников Марат, 7 класс
Кулаковский Даниил, 9 класс
Петров Андрей, 11 класс
28. Команда «ЯКУТСК-10», МОБУ НПСОШ2, руководитель Петров Айсен
 Алексеевич
Степанов Элэй Олегович, 9 класс
Акимов Альберт Анатольевич, 9 класс
Николаев Павел, 9 класс

УСЛОВИЯ ЗАДАЧ

А. Зимовье

Толлуман — фермер и держит K коров. Для того, чтобы перезимовать, одной корове необходимо L тонн сена. Толлуман за лето накопил M тонн сена. Определите, хватит ли запасенного сена на всю зиму. Если не хватит, то сколько Толлуману нужно дополнительно закупить сена?

Формат входных данных

На вход вашей программе дается три целых числа — K, L, M ($0 < K, L, M \leq 10^{10}$).

Формат выходных данных

Выведите слово «YES», если запасенного сена хватит. Если не хватает, через пробел выведите слово «NO» и количество дополнительно необходимого сена.

Примеры входного и выходного файлов

стандартный ввод	стандартный вывод
2 10 21	YES
3 5 10	NO 5

В. Победитель

Айтал, Дьулуур и Моххой — три закадычных друга, которые при встрече всегда соревнуются между собой. Друзья перепробовали абсолютно все игры. Сегодня они решили посоревноваться, у кого больше денег в кармане. Игру выигрывает тот, у кого больше всех денег сейчас. Победитель в игре может быть только один, во всех остальных случаях объявляется ничья. Зная, сколько денег в кармане у каждого из друзей, ответьте, кто же будет победителем в сегодняшней игре друзей.

Формат входных данных

На вход вашей программе даются три целых числа A, B и C — количество рублей в кармане у Айтала, Дьулуура и Моххой соответственно ($0 < A, B, C \leq 10^{10}$).

Формат выходных данных

Ваша программа должна вывести «А», если победителем игры является Айтал, «D» — если Дьулуур, «М» — если Моххой, «Т» — если победителя определить не удастся.

Примеры входного и выходного файлов

стандартный ввод	стандартный вывод
10 2 20	M
335 335 335	T

С. RAID 5

Имеется сообщение из N символов. Каждый символ в нем имеет свой числовой код. Допустимыми символами являются:

- цифры (код совпадает со значением цифры);
- строчная буква латинского алфавита (коды имеют значения в диапазоне 10–35);
- заглавная буква латинского алфавита (коды имеют значения в диапазоне 36–61);
- подчеркивание (код символа 62);
- точка (код символа 63).

Сообщение разбивается на K блоков по следующему принципу: первый символ записывается в первый блок, второй символ записывается во второй блок и т. д. $(K+1)$ -й символ снова записывается в первый блок, $(K+2)$ -й символ — во второй блок и т. д.

Все блоки должны иметь одинаковое количество символов, обозначаемое как L . Если в блоке не хватает символа, тогда в него следует дописать символ «точка» (с кодом 63).

Например, сообщение «Hello_world» разделяется на 3 блока следующим образом: «Hlwl», «eood» и «l_r.».

Для повышения надежности к блокам сообщения добавляется ещё один блок такой же длины. Каждый символ в добавочном блоке подбирается таким образом, чтобы сумма кодов всех символов на i -й позиции в разных блоках была кратна 64.

То есть, дополнительный блок для предыдущего примера равен «01jv». Если сложить коды первых символов в каждом блоке, получим: $43(H) + 14(e) + 21(l) + 50(O) = 128$. Если сложить коды вторых символов в каждом блоке, получим: $21(l) + 24(o) + 62(_) + 21(l) = 128$.

В процессе передачи блоков сообщения один из них был утерян. Напишите программу, которая сможет восстановить его.

Формат входных данных

В первой строке входных данных заданы K и L — целые положительные числа, разделенные пробелом ($0 \leq K, L \leq 10^2$).

В следующих K строках заданы известные блоки сообщения.

Формат выходных данных

Выведите символы недостающего блока сообщения.

Пример входного и выходного файлов

стандартный ввод	стандартный вывод
3 4 Hlwl eood Oljv	l_r.

D. Складской учет

Уйгуна — молодая начинающая предпринимательца, которая недавно открыла магазин. Каждую неделю она должна проводить аудит остатков по товарам и составлять список товаров на пополнение. Если остаток товара в магазине меньше A единиц, то он включается в список на пополнение остатков. Помогите Уйгуне автоматизировать данную операцию, разработайте программу для составления списка товаров.

Формат входных данных

На первой строке заданы два целых числа A и N , минимальный остаток и количество видов товаров в магазине ($0 < A, N \leq 10^5$). В каждой из следующих N строк заданы: разделенные пробелом наименование товара и его остаток на данный момент a_i ($0 < a_i \leq 10^{10}$).

Формат выходных данных

Ваша программа должна вывести список наименований товаров, остатки по которым необходимо пополнить на этой неделе, каждое с новой строки.

Пример входного и выходного файлов

стандартный ввод	стандартный вывод
10 5 Bread 20 Butter 5 Sausages 8 Soup 15 Shower Gel 4	Butter Sausages Shower Gel

Е. Призеры универсиады

Соревнования по тяжёлой атлетике в Российской Федерации проводятся в следующих видах программы — рывок, толчок и двоеборье (рывок и толчок). Факторами, учитываемыми для классификации спортсменов в сумме двоеборья являются:

1. Лучший результат: наибольший результат даёт первое место; если у спортсменов они идентичные, тогда учитываются ещё:
2. Лучший результат в толчке: наименьший результат даёт первое место; если они идентичные, тогда учитываются ещё:
3. Номер попытки, в которой показан лучший результат в толчке: наименьший номер попытки даёт первое место; если они идентичные, тогда учитываются ещё:
4. Предыдущая попытка (попытки): наименьший вес в попытке даёт первое место; если они идентичные, тогда учитывается ещё:
5. Номер жеребьёвки: меньший номер — первый.

У юного программиста Васи старший брат Петя — участник соревнований Всероссийской универсиады в этом виде спорта. Он еще до объявления итогов прислал Васе протокол соревнований своей весовой категории и попросил определить тройку призеров. Вася решил составить программу, но правила соревнований оказались слишком запутанными для него. А вы смогли бы ему помочь?

Формат входных данных

Васе удалось немного обработать рабочий протокол, и теперь первая строка входных данных содержит натуральное число N — количество участников соревнований ($1 \leq N \leq 1000$). Далее следует N строк, каждая из которых пред-

ставляет результаты выступления отдельного спортсмена в следующем формате:

T *University Name Surname* $J_1 J_2 J_3 P_1 P_2 P_3$

Здесь:

- T — номер жеребьёвки, натуральное число ($1 \leq T \leq N$);
- *University, Name, Surname* — соответственно, вуз, за который выступает спортсмен, его имя и фамилия — строки из символов латинского алфавита, каждая из которых не превышает по длине 80 символов;
- $J_1 J_2 J_3$ — результаты трех попыток в рывке в килограммах, неотрицательные целые числа ($0 \leq J_1, J_2, J_3 \leq 130$), 0 означает, что попытка не увенчалась успехом и заявленный вес не был взят;
- $P_1 P_2 P_3$ — аналогично, результаты трех попыток в толчке.

Все параметры в строке разделены ровно одним пробелом, ведущие и хвостовые пробелы отсутствуют.

Формат выходных данных

Построчно выведите тройку победителей (1, 2, 3 место) согласно следующему формату (между словами — ровно один пробел):

University Name Surname

Спортсменов без зачетных попыток хотя бы в одном виде следует проигнорировать. Если победителей 1 или 2, следует вывести только их. Если невозможно выявить ни одного победителя, то следует вывести строку «No results».

Примеры входного и выходного файлов

стандартный ввод	стандартный вывод
8	NEFU Lev Rykov
6 MSU Igor Permyakov 85 0 100 0 0 100	RSPU Igor Menkov
7 NSU Ivan Belyaev 0 100 0 0 120 0	NSU Ivan Belyaev
1 NEFU Lev Rykov 90 95 105 114 115 0	
2 MAI Sergey Ivanov 85 94 105 0 0 0	
5 HSE Petr Popov 0 100 0 110 115 120	
4 RSPU Igor Menkov 100 0 0 120 0 0	
3 NEFU Duolan Uyurov 90 0 95 110 0 0	
8 MPSU Alexey Vetrov 0 0 0 0 0 0	

стандартный ввод	стандартный вывод
3 2 MSU Ivan Smirnov 85 94 0 79 84 91 1 MAI Evgeny Kuznetsov 0 0 0 0 0 0 3 NEFU Luka Martynov 94 0 0 80 0 91	MSU Ivan Smirnov NEFU Luka Martynov
2 1 NSU Fedor Krylov 0 0 0 0 0 0 2 MSU Fedor Krukov 0 0 0 0 0 0	No results

Г. Философский камень

Никола Фламель в этом году нашел тайную книгу заклинаний знаменитого чернокнижника Италии. Эта книга замечательна тем, что в ней содержится рецепт получения философского камня. Но этот рецепт будет иметь силу, только если он приготовлен во время великого противостояния планет (когда все планеты системы выстраиваются в одну линию). Если известны периоды обращения планет (в земных днях), а также то, что последнее великое противостояние планет было K дней тому назад, ответьте, когда Никола Фламель сможет получить философский камень.

Формат входных данных

На вход программе дается последовательность целых чисел. Первая строка содержит числа N и K — количество планет и дней, прошедших с последнего великого противостояния планет ($0 < N \leq 20$, $1 \leq K \leq 10^3$). На следующей строке заданы разделенные пробелом периоды обращения планет ($1 \leq A_i < 10^2$, $i = 1, \dots, N$) в днях.

Формат выходных данных

Ваша программа должна вывести одно единственное целое число — наименьшее количество дней, через которые Никола Фламель сможет получить философский камень.

Пример входного и выходного файлов

стандартный ввод	стандартный вывод
3 100 12 22 34	2144

Г. Номера квартир

Лёлек и Болек работают на стройке. На завершающей стадии строительства нового многоквартирного дома им поручили нанести краской на дверях квартир номера. В первой квартире дома расположится детский сад, на дверях которого будет красивая табличка, так что туда решили ничего не наносить. Бумажные трафареты для краски одноразовые, и каждый содержит ровно одну цифру, так что нужно заранее точно посчитать их количество. Зная число квартир N в доме, определите, сколько всего трафаретов с каждой из цифр потребуется? Лёлек и Болек – шустрые ребята, и могут успеть много. Поэтому выведите не сами величины, а их остатки от деления на 10^9 .

Формат входных данных

В первой строке входного файла содержится одно натуральное число N ($1 \leq N \leq 10^{1000}$).

Формат выходных данных

В выходной файл вывести через пробел десять чисел M_0, M_1, \dots, M_9 , где M_k — остаток от деления числа появлений цифры k в нумерации квартир на 10^9 , $k = 0, 1, \dots, 9$.

Пример входного и выходного файлов

стандартный ввод	стандартный вывод
11	1 3 1 1 1 1 1 1 1 1

Н. Путь Флипа

Кузнечик Флип и пчелка Майя — лучшие друзья. К великому огорчению Флипа в этом году их дома оказались разделены многополосной автострадой. Какое дело людям до дружбы каких-то там насекомых! За все время строительства только Майя прилетала к Флипу, но сегодня у нее день рождения, и пчелка принимает гостей у себя дома.

К счастью, на границах полос автострады довольно много островков безопасности, и кузнечик рассчитывает перебраться через автостраду, перепрыгивая в целях безопасности ровно через одну полосу от островка к островку. Непонятно, чем руководствовались те, кто делал разметку, но количество островков на границах разное, да и нарисовали их безо всякой логики — то близко друг от друга, то далеко.

Помогите Флипу перебраться через автостраду — подскажите, на какие островки нужно прыгать, чтобы по длине весь путь кузнечика был минимальным. Известно, что дом Флипа находится на координатной плоскости в точке $(0, 0)$, границы автострады параллельны оси x , первый прыжок кузнечик делает на внешнюю границу автострады со стороны своего дома, а последний — с внешней границы автострады со стороны дома Майи.

Формат входных данных

В первой строке входных данных записаны три целых числа: a и b — соответственно, кратчайшие расстояния от домов Флипа и Майи до автострады ($1 \leq a, b \leq 100$), m — координата дома Майи по оси x ($-1000 \leq m \leq 1000$).

Во второй строке заданы два целых числа n ($1 \leq n \leq 100$) — количество полос автострады и h — ширина каждой полосы ($1 \leq h \leq 10$). В следующих $n - 1$ строках задано $k_i + 1$ чисел: первое натуральное число k_i — количество островков безопасности на i -й границе полосы ($1 \leq k_i \leq 20$), остальные числа — координаты островков по оси x , все координаты целые и не превышают по модулю 1000.

Формат выходных данных

Выведите построчно x -координаты точек приземлений кузнечика по минимальному пути с точностью не хуже 10^{-4} .

Если вариантов ответа более одного, выведите любой из них.

Пример входного и выходного файлов

стандартный ввод	стандартный вывод
2 2 3	0.5000
2 2	1.0000
3 -4 1 6	2.0000

Комментарии

y -координата дома Майи больше 0.

I. Автобусные остановки

Один недовольный житель города N опубликовал в Инстаграме стори о том, как он ежедневно добирается до работы на автобусе, делая по две пересадки. Депутаты специальной комиссии городской думы, после тщательного разбирательства, пришли к выводу, что схема движения городских автобусов неэффективна.

Чтобы решить эту проблему, мэр постановил создать новый автобусный маршрут, который должен будет связать самые неудачные остановки. В своем Твиттере он объявил конкурс, победителем которого станет человек, нашедший наибольшее количество пар неудачно расположенных остановок.

Чтобы разъяснить, что же на самом деле имел в виду мэр, в официальной группе мэрии города N ВКонтакте был опубликовано следующее коммюнике:

Расстоянием между двумя остановками будем называть минимальное возможное количество межостановочных интервалов, которое необходимо проехать по пути от одной из них до другой.

Пару остановок назовем *неудачной*, если невозможно найти другую пару остановок с большим расстоянием, чем у неё.

Победителем конкурса становится участник, который быстрее остальных сможет найти все пары неудачных остановок.

Чтобы помочь мэрии быстро оценить все приходящие решения, нужно написать программу, которая определит количество пар неудачных остановок и расстояние между ними.

Формат входных данных

В первой строке входных данных дается одно целое положительное число N — количество автобусных маршрутов ($1 \leq N \leq 10^2$).

Во второй строке входных данных дается N целых положительных чисел K_i — количество остановок на каждом из маршрутов ($1 \leq K_i \leq 10^4$).

В последующих N строках задано по K_i целых положительных чисел v_{ij} — номера автобусных остановок, составляющих i -й маршрут ($1 \leq v_{ij} \leq 10^4$).

Формат выходных данных

В качестве результата ваша программа должна вывести два целых положительных числа, разделенных пробелом — количество пар неудачных остановок и расстояние между ними.

Пример входного и выходного файлов

стандартный ввод	стандартный вывод
3 4 2 4 1 2 3 4 3 6 5 6 7 8	5 1

РЕШЕНИЯ

А. Зимовье

Простая задача на ветвления. Пример решения на языке программирования C++ приведен ниже:

```
#include<iostream>
using namespace std;

void main(){
    int k,l,m;
    cin >> k >> l >> m;
    if (k*l<=m)
        cout << "YES";
    else
        cout << "NO " << k * l - m;
}
```

В. Победитель

Задача на множественное ветвление. Надо проверить условия проверки выигрыша отдельных героев задачи, а иначе вывести «Т». Пример решения задачи на языке программирования C++ приведен ниже:

```
#include<iostream>
using namespace std;
void main(){
    long long a,d,m;
    cin >> a >> d >> m;
    if (a > d && a > m)
        cout << 'A';
    else if(d > a && d > m)
        cout << 'D';
    else if(m > a && m > d)
        cout << 'M';
    else
        cout << 'T';
}
```

С. RAID 5

Для упрощения решения создадим алфавит — одномерный массив, который будет содержать все возможные символы из сообщения. В дальнейшем, чтобы определить код символа, будем искать его в алфавите. Код символа будет совпадать с его индексом (положением).

Организуем цикл который обходит символы в блоках. Для каждого символа суммируем коды из всех блоков. Затем, используя операцию взятия остатка от деления можно получить код соответствующего символа из недостающего блока:

$$r_i = (64 - s_i) \% 64$$

где s_i — сумма кодов i -го символа из всех блоков, r_i — искомый код символа из недостающего блока, и $\%$ — оператор остатка от деления.

На языке Python после ввода данных решение может иметь вид:

```
alphabet = '0123456789'
alphabet += 'abcdefghijklmnopqrstuvwxyz'
alphabet += 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'
alphabet += '._'

answer = ''
for i in range(L):
    code = 0
    for j in range(K):
        block = blocks[j]
        symbol = block[i]
        code += alphabet.find(symbol)
    answer += alphabet[(64 - code) % 64]

print(answer)
```

D. Складской учет

По мере считывания данных из строки следует выделить название товара и его остаток на данный момент. Если остаток меньше заданной величины A , то выводим название товара.

Определенную трудность для некоторых языков программирования может представлять вычленение названия товара и его остатка. Для этого просматри-

ваем считанную строку с конца к началу и надо запомнить позицию с которого начинаются цифры. Все, что находится левее этой позиции это название товара, а правее это его остаток.

Пример решения данной задачи на языке программирования Python:

```
a, n = map(int, input().split())
for i in range(n):
    data = input().split()
    o = int(data[-1])
    if o <= a:
        print(' '.join(data[:-1]))
```

Е. Призеры универсиады

Ограничения задачи позволяют отсортировать весь список спортсменов в порядке «от лучших к худшим» и определить тройку призеров. Хотя, как известно, можно реализовать поиск трех лучших спортсменов и за $O(N)$.

Сложность задачи заключается в том, чтобы суметь сравнить результаты двух спортсменов и выбрать лучшего.

Для большей определенности при чтении данных «проигнорируем» тех спортсменов, у которых не оказалось зачетных результатов хотя бы по одному виду, то есть тех, у кого либо все попытки в рывке провалены, либо все в толчке, либо в обоих сразу. Если при этом ни одного спортсмена не осталось, выводим «No results». Параллельно создаем три вспомогательных списка: список данных для вывода (вуз, имя и фамилия), список данных для обработки и список номеров спортсменов. Для того, чтобы упростить обмены, будем представлять именно номера спортсменов, а не все их данные. В список обрабатываемых данных для каждого спортсмена сохраним: сумму лучших результатов по обоим видам, лучший результат в толчке, номер попытки в толчке, результаты в предыдущей и предпредыдущей попытках (если таковых не окажется, зададим их нулями), а также номер жеребьевки.

В авторском решении данной задачи была использована «гномья» сортировка, для сравнения результатов спортсменов была составлена специальная функция *better*. Данная функция скрупулезно реализует описанные в условии задачи правила.

А именно сравнивает сначала суммы лучших результатов по каждому из видов, при их равенстве — лучшие результаты в толчке. При этом выше ранжиру-

ется спортсмен, у которого данный результат МЕНЬШЕ (видимо, рывок является более сложным упражнением).

Далее, при равенстве результатов в толчке, рассматриваются номера попыток. Выше ранжируется спортсмен, который взял вес с более ранней попытки. Если и в этом они оказались равны, переходит к сравнению результатов в более ранних попытках (при их наличии). И здесь, если один из спортсменов вообще не взял никакой вес, он ПРОИГРЫВАЕТ. Если же все попытки толчка у обоих оказались одинаковы, то их судьбу решает жребий — выше ранжируется тот спортсмен, который вышел на помост раньше.

Для интересующихся приведем фрагмент программы на языке Python (сортировка и вывод результатов):

```
#гномья сортировка, m - список номеров спортсменов
i=k-1
while i>0:
    if better(m[i], m[i-1]):
        tmp = m[i]
        m[i]= m[i-1]
        m[i-1] = tmp
        if i <k-1:
            i += 1
    else:
        i -= 1

for i in range(min(3, k)):
    print(athletes[m[i]])
```

Г. Философский камень

Задача сводится к нахождению НОК($a_1, a_2, a_3, \dots, a_n$). Использование перебора для нахождения НОК множества чисел не подходит для в этого случая. Для решения задачи воспользуемся теоремой, согласно которой НОК последовательности чисел можно вычислить, рекурсивно вычисляя НОК двух чисел:

$$m_1 = \text{НОК}(a_1, a_2), m_2 = \text{НОК}(m_1, a_3), \dots, m_k = \text{НОК}(m_{k-1}, a_n)$$

тогда

$$m_k = \text{НОК}(a_1, a_2, a_3, \dots, a_n).$$

Для нахождения НОК двух чисел необходимо воспользоваться алгоритмом Евклида.

Г. Номера квартир

Подсчитаем, сколько раз в номерах квартир определенная цифра k появляется в какой-нибудь заданной позиции p . Эта величина зависит от цифр самого числа $N = d_p \dots d_2 d_1$. Начнем рассмотрение с примера.

Пусть $N = 742\bar{6}310$, и нас интересует (подчеркнута) четвертая справа позиция. Количество вхождений во все числа от 1 до N в этой позиции $p = 4$ будет различным для разных цифр k . Будем отдельно интересоваться, что может быть записано слева от данной позиции, обозначая эту часть ???, и справа, обозначая правую часть звездочками. Сведем все случаи в таблицу:

Цифра k	0	$1 \leq k < 6$	$k > 6$	$k = 6$
Входит	от $1\bar{0}***$	от $\bar{3}***$	от $\bar{8}***$	от $\bar{6}***$
в числа	до $742\bar{0}***$	до $742\bar{3}***$	до $741\bar{8}***$	до $741\bar{6}***$
				и от 7426000 до $742\bar{6}310$

Выделим числа L_p и R_p , образованные цифрами числа N до и после выбранной позиции $p = 4$:

$$N = \begin{array}{|c|c|c|} \hline 742 & \bar{6} & 310 \\ \hline L_4 & d_4 & R_4 \\ \hline \end{array}$$

Вместо каждой из звездочек справа может стоять любая цифра, то есть вариантов заполнения звездочек $10^3 = 10^{p-1}$. Варианты заполнения вопросительных знаков слева зависят от числа $L_4 = 742$. Кроме того, если слева стоит само L_p , а в выбранной позиции цифра $d_p = \bar{6}$, то продолжить такое число справа можно только записями (с ведущими нулями) чисел от 0 до $R_4 = 310$.

Обозначим через $m_p(k)$ общее число вхождений цифры k в p -ю позицию в числа от 1 до N . Обобщая проведенное рассмотрение, получаем, что для $1 < p < P$:

$$m_p(k) = \begin{cases} L_p \times 10^{p-1}, & k = 0 \\ (L_p + 1) \times 10^{p-1}, & k < d_p, \\ L_p \times 10^{p-1} + (R_p + 1), & k = d_p, \\ L_p \times 10^{p-1}, & k > d_p \end{cases}$$

Доработав эту формулу для крайних разрядов, и просуммировав для каждой цифры по всем позициям, а также не забыв вычесть единицу для первой квартиры, получим решение. Все вычисления следует проводить по модулю 10^9 , поэтому при $p > 1$ ненулевыми будут только вклады от слагаемых вида $R_p + 1$.

Н. Путь Флипа

Данная задача сочетает в себе стандартное двумерное динамическое программирование и элементы вычислительной геометрии.

Целевая функция будет зависеть от двух параметров: номера границы i и номера островка безопасности j и иметь своим значением минимальное расстояние пути кузнечика из начала координат в точку с координатами (i, j) .

Кроме этого, для восстановления маршрута минимальной длины понадобится вторая целевая функция от тех же параметров, имеющая своим значением номера точек на предыдущей границе, прыжок из которых в точку (i, j) обеспечивает минимальный путь.

Чтобы вычислить j -е координаты точек, в которых кузнечик приземляется на внешних границах дороги со стороны своего дома и со стороны дома пчелки Майи, воспользовавшись подобием треугольников, получим соответственно, формулы:

$$x[0][j] = a * x[1][j] / (a + h), x[n][jmin] = x[n-1][jmin] - (x[n-1][jmin] - m) * h / (b + h).$$

Решение задачи выполняется в два этапа: вычисление значений целевых функций от 1-й границы между полосами автостреды до последней, затем определяется номер точки, на предпоследней границе, через которую проходит минимальный путь кузнечика и восстанавливается его маршрут с использованием второй целевой функции в обратном порядке.

I. Автобусные остановки

Задача основана на поиске кратчайшего пути в графе. Ограничения на входные данные позволяют использовать стандартные алгоритмы: поиск в ширину, алгоритм Дейкстры или алгоритм Флойда-Уоршелла.

В простейшем случае, после чтения входных данных следует вычислить все пары расстояний между остановками. Затем определить максимальное расстояние и соответствующее количество вершин графа.

Задача также предоставляет широкое поле для оптимизации. Трудоемкость алгоритма можно значительно снизить следующими способами:

- Рассматривать только конечные остановки автобусных маршрутов.
- Изначальный невзвешенный граф можно превратить во взвешенный, приписав ребрам вес равный 1. Затем, можно объединить промежуточные остановки, сложив веса объединяемых ребер.

ИТОГИ XVII РЕСПУБЛИКАНСКОЙ КОМАНДНОЙ ОЛИМПИАДЫ ШКОЛЬНИКОВ ПО ПРОГРАММИРОВАНИЮ

№	Команда	A	B	C	D	E	F	G	H	I	Задачи	Время	МЕСТО	ДИПЛОМ
1	ЯКУТСК-1 (Крыльков Максим Игоревич, Захаров Айтал Анатольевич, Павловский Владислав Витальевич)	+1 0:06	+ 0:11	+ 0:29	+1 0:36	+4 4:14	+	+	·	·	7	684	1	I степени
2	ЭРЭЛ-1 (Захаров Алексей Викторович, Зыков Тимур Алексеевич, Семенов Никита Сергеевич)	+1 0:04	+ 0:06	+1 1:27	+ 0:12	+3 4:15	+1 0:26	-3	-4	·	6	510	2	I степени
3	ЯКУТСК-4 (Колодезников Егор Иннокентьевич, Туприн Айтал Константинович, Петров Тимур Робертович)	+	+	+1 0:12	+	·	+	·	·	·	5	216	3	II степени
4	ЯКУТСК-2 (Лаптев Максим Сергеевич, Николаев Роберт Михайлович, Хуанг Ян Мингцзиевич)	+	+	+1 0:08	+	·	+	-1 1:02	·	·	5	230	4	II степени
5	СПТЛП-1 (Дмитриев Аркадий Валериевич, Захарова Марина Михайловна, Фролова Ксения Георгиевна)	+	+	+3 0:31	+	·	+	·	·	·	5	306	5	II степени
6	НЕРЮНГРИ (Гуреев Никита Андреевич, Сухих Дмитрий Сергеевич, Амвросов Артем Игоревич)	+	+	+2 0:13	+	-5 0:27	+5 0:59	·	-5	·	5	360	6	II степени
7	ЭРЭЛ-2 (Замятин Андрей Иванович, Соров Айаал Львович, Жирков Айал Андреевич)	+	+1 0:05	+5 0:22	+	·	+	-1 1:24	·	·	5	434	7	II степени
8	ОКТЕМЦЫ-1 (Корякин Сергей Сергеевич, Ксенофонтов Василий Ильич, Терентьев Данил Степанович)	+	+	+	+3 0:11	·	+1 2:53	-1	·	·	5	467	8	III степени
9	ЭРЭЛ-3 (Павлов Никифор Михайлович, Кузьмин Андрей Алексеевич, Уваровская Лена Александровна)	+	+	+4 0:12	+	·	+1 1:23	·	·	·	5	482	9	III степени
10	ЯКУТСК-3 (Поскачин Владимир, Федорова Ева, Кангелдиева Асель)	+1 0:08	+1 0:13	+	+2 3:12	·	+3 2:40	·	·	·	5	558	10	III степени
11	НЮРБА (Иванов Александр Прокопьевич, Семёнов Ефим Николаевич, Ефремов Вадим Афанасьевич)	+	+	+4 0:09	+	·	+1 3:32	·	·	·	5	583	11	III степени
12	ЯКУТСК-9 (Местников Марат Павлович, Кулаковский Даниил Семенович, Петров Андрей Николаевич)	+	+2 0:07	+	+2 0:54	-18 4:30	+14 4:30	-1	·	·	5	692	12	III степени
13	ОКТЕМЦЫ-2 (Парников Василий Васильевич, Егорова Дайаана Андреевна, Васильева Анна Петровна)	+	+	-2 0:09	+1 0:30	·	+	·	·	·	4	120	13	
14	ЧУРАПЧА-1 (Илларионов Артем Иванович, Иванов Радомир Антонович, Куличкин Тимур Фёдорович)	+	+	+	+2 0:14	·	-5 0:42	-5	·	·	4	239	14	

[illegible]