

1. Метелики (but.*)

Петрик наловив у бабусі в садочку N метеликів, кожен з яких має свою назву a,b,c,...z. Усі метелики дуже красиві, але Петрику найбільше подобаються метелики a, b та c. Допоможіть Петрику порахувати кількість метеликів кожного з найкрасивіших видів.

Вхідні дані

Перший рядок вхідного файлу містить число N – кількість метеликів, яку спіймав Петрик. Другий рядок містить N маленьких латинських літер – назви метеликів, які здавив Петрик.

Вихідні дані

У вихідний файл потрібно вивести три числа: кількість метеликів виду a, b та c, які упіймав Петрик.

Приклад даних:

but.in	but.out
8 cbbabdaz	1 3 1

2. Модернізація у чаклунському магазині братів Уїзлі (card1.* або card2.*)

Брати Фред та Джордж Уїзлі виріши зробити свій магазин чаклунських приколів дещо більш сучасним та відкрити філіал у нечаклунському світі. Але ж вони дізналися, що дуже багато людей звикли розраховуватись у магазинах не золотими галеонами, а кредитними картками. До відкриття нового магазину залишились лічені години, і брати Уїзлі вже не встигають замовити сучасні термінали для роботи з кредитними картками. Їм довелося вигадувати спеціальний чаклунський термінал. Все було б добре, але термінал не вміє перевіряти достовірність кредитної картки, як це роблять звичайні термінали ще до з'єднання з банком. Вам потрібно написати програму, що за номером кредитної картки, наданої покупцем магазину, визначить, чи є вона дійсною.

Кредитні картки є в дуже великій кількості людей в усьому світі, тому їх номери мають бути досить великими: номери карток American Express складаються з 15 цифр, MasterCard - з 16 цифр, а картки Visa можуть мати номери з 13 або 16 цифр. Усі цифри десяткові від 0 до 9. Але це не всі умови, яким мають задовільняти номери кредитних карток. Номери карток American Express починаються з 34 або 37, номери MasterCard починаються з 51, 52, 53, 54 або 55, а усі номери на картках Visa починаються з 4.

Крім того, номери на усіх кредитних картках мають ще й додаткову "контрольну суму" (математичне співвідношення між принаймні однією цифрою та іншими). Це дозволяє визначити, чи не припустилася людина помилок, уводячи номер кредитної картки. Алгоритм визначення контрольної суми картки був запропонований працівником компанії IBM Гансом Пітером Луном у 1960 році і досі застосовується для банківських карток:

1. Помножте на 2 кожен другу цифру номера картки, починаючи з другої з правого кінця номера цифри і до лівого кінця номера. Визначте суму усіх цифр у результатах множення.
2. Додайте отриману на першому кроці суму до суми цифр, що не були помножені на 2.
3. Якщо остання цифра у отриманому результаті 0, то номер вважається дійсним.

Ця задача має бути розв'язана в одному з двох варіантів. Визначтесь з тим, який варіант для вас більш прийнятний і здавайте лише його. Врахуйте, що перший варіант розв'язку може принести не більше 50% можливих балів.

Варіант 1:

Назвіть програму **card1.pas (.c)**.

Ім'я файлу з вхідними даними: card.in.

Ім'я файлу з результатом: card.out

Вхідні дані: У першому рядку вхідного файлу записано кількість цифр у номері картки. Другий рядок

містить послідовність цифр номеру картки, розділених пропусками.

Вихідні дані: У єдиному рядку вивести одне з повідомлень в залежності від виду кредитної картки AMEX, MASTERCARD, VISA або ERROR, якщо номер картки не дійсний.

Приклад даних:

№	card.in	card.out
1	15 3 7 8 2 8 2 2 4 6 3 1 0 0 0 5	AMEX
2	10 7 7 2 2 5 7 4 5 0 1	ERROR

Варіант 2:

Назвіть програму **card2.pas (.c)**.

Ім'я файлу з вхідними даними: card.in.

Ім'я файлу з результатом: card.out

Вхідні дані: У першому рядку вхідного файлу записаний номер кредитної картки.

Вихідні дані: В єдиному рядку вивести одне з повідомлень в залежності від виду кредитної картки AMEX, MASTERCARD, VISA або ERROR, якщо номер картки не дійсний.

Приклад даних:

№	card.in	card.out
1	378282246310005	AMEX
2	7722574501	ERROR

3. Петрик П'яточкін пише олімпіаду (olympiad.*)

Прочитавши чимало книжок про алгоритми, Петрик П'яточкін прийшов на районну олімпіаду з інформатики. Перед початком олімпіади з'ясувалося, що багато хто з учасників знає одне одного. Тож організатори вирішили убезпечитися й розсадити всіх учасників по двох кабінетах, де проходить олімпіада, у такий спосіб, щоб жодні два знайомі між собою учасники не сиділи в одному кабінеті. Напишіть програму, яка допоможе організаторам зробити це або повідомить, що розсадити в такий спосіб учасників неможливо.

olympiad.in	olympiad.out
5 6	1 2 4
2 5	3 5
2 3	
1 5	
4 5	
3 4	
1 3	
3 3	0
1 2	0
2 3	
1 3	

Вхідні дані

У першому рядку вхідного файлу вказано два натуральних числа n та m — кількість учасників районної олімпіади та кількість пар знайомих учасників, $2 \leq n < 2000$, $m < 450000$. У кожному з наступних m рядків задано по два числа a_j, b_j — номери знайомих

між собою учасників, $1 \leq a_j, b_j \leq n$, $1 \leq j \leq m$. Жодна пара номерів a_j, b_j не повторюється у вхідному файлі більше ніж один раз. Крім того, вхідні дані гарантують, що є не більше ніж один спосіб розсадити учасників по двох кабінетах, щоб жодні два знайомі не сиділи в одному кабінеті.

Вихідні дані

У першому рядку вихідного файлу виведіть у порядку зростання номери учасників, які мають сидіти в кабінеті разом із учасником під номером 1 (Петриком П'яточкіним) включно із самим числом 1 (Петриком). У другому рядку виведіть у порядку зростання номери учасників, які повинні сидіти в іншому кабінеті.

Якщо жодне розташування учасників не задовольняє умову задачі, в обох рядках виведіть по нулю.