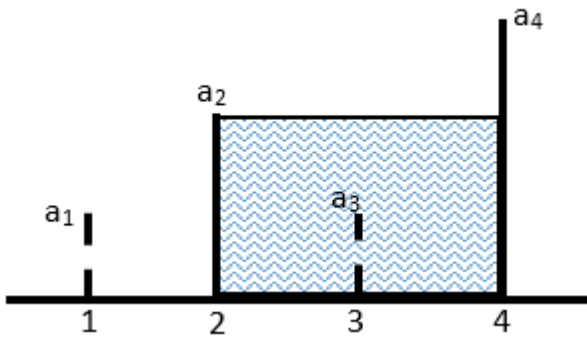


Задача Затоплення (*flooding*).

Герой відомого мультсеріала Коливан вирішив побудувати собі басейн. Оскільки він дуже скупий, він намагається використати будівлю, що вже існує.



Будівля являє собою абсолютно рівний коридор одиничної ширини, в якому є N перегородок. Якщо розмістити вісь X вздовж коридору, усі перегородки будуть знаходитись точно в її цілочисельних координатах з кроком 1, причому по ширині перегородки займають весь коридор, а висоти можуть відрізнятися. Басейн, що створюється, повинен мати максимально великий можливий об'єм за умови, що з усіх існуючих перегородок потрібно залишити *тільки* дві та збільшувати їх висоту заборонено.

Формат введення/виведення.

Програма *flooding* зчитує з клавіатури (стандартного пристрою введення) ціле число N ($2 \leq N \leq 10^5$) - кількість перегородок. Потім зчитується N цілих чисел a_i ($1 \leq a_i \leq 10^9$) - висоти перегородок.

Програма *flooding* виводить на екран (стандартний пристрій виведення) єдине число - максимально можливий об'єм створюваного басейна з урахуванням вказаних обмежень .

Приклад вхідних та вихідних даних.

Введення	Виведення
4 1 2 1 3	4

Задача Фазенда (*hacienda*).

Для своєї приятельки баби Яги Коливан вирішив побудувати фазенду в лісі. Баба Яга дуже любить трикутники, а тому Коливан хоче, щоб ділянка для фазенди була трикутної форми з деревами у вершинах. Оскільки баба Яга не хоче шкодити природі та рубати ліс, Коливану необхідно знайти такі три дерева, щоб всередині огорожі не було інших дерев.

Щоб мати можливість вибору, Коливану потрібно знати, скільки всього в лісі існує таких ділянок.

Формат введення/виведення.

Програма *hacienda* зчитує з клавіатури (стандартного пристрою введення) ціле число N ($3 \leq N \leq 200$) - кількість дерев у лісі. Потім зчитується N пар цілих чисел x_i, y_i ($|x_i|, |y_i| \leq 10^5$) - координати дерев у лісі, причому жодні три дерева не знаходяться на одній прямій.

Програма **hacienda** виводить на екран (стандартний пристрій виведення) єдине число - кількість трикутних ділянок, які не містять жодного дерева.

Приклад вхідних та вихідних даних.

<i>Введення</i>	<i>Виведення</i>
5	8
-9 -10	
0 4	
8 8	
-1 10	
1 -7	

Задача Хитрий паліндром (*pal333*).

Після вибору ділянки та розробки проектів будівництва басейну та фазенди Коливан отримав кошторис на виконання будівельних робіт. З урахуванням зміни курсу долара сума вийшла дуже великою і Коливан, щоб заспокоїтися, вирішив визначити, чи є ця сума паліндромом, тобто таким числом, яке читається в обох напрямках однаково. Потім Коливан поставив питання, скільки існує способів отримати паліндром із заданої послідовності цифр, вилучаючи з неї деяку, можливо порожню, множину цифр. Нарешті, він вирішив ще більше ускладнити собі задачу та знайти серед них всі числові паліндроми, що діляться націло на число 333. Допоможіть йому вирішити цю задачу.

Формат введення/виведення.

Програма **pal333** зчитує з клавіатури (стандартного пристрою введення) ціле число N ($1 \leq N \leq 100$) - кількість цифр числа. Потім зчитуються N цифр.

Програма **pal333** виводить на екран (стандартний пристрій виведення) єдине число - кількість способів отримання чисел-паліндромів, що діляться націло на число 333, із заданої послідовності цифр. Відповідь повинна бути виведена по модулю 10007. Паліндром може починатися з ведучих нулів.

Приклад вхідних та вихідних даних.

<i>Введення</i>	<i>Виведення</i>
6	8
303330	

Пояснення до прикладу: із заданої послідовності цифр можна отримати такі паліндроми, що діляться націло на 333: 0 (1-й нуль); 0 (2-й нуль); 00; 333 (1-ша, 2-га та 3-тя трійки); 333 (1-ша, 2-га та 4-та трійки); 333 (1-ша, 3-тя та 4-та трійки); 333 (2-га, 3-тя та 4-та трійки); 03330.