

Завдання для учнів 9-11 класів
Час виконання – 5 годин

Задача Table2021 Дано прямокутну таблицю A розміром $n \times n$ ($1 \leq n \leq 100$), що заповнена цілими числами. ($-65000 \leq A_{i,j} \leq 65000$) Потрібно отримати прямокутну таблицю B розміром $n \times n$, кожний елемент якої ($b_{i,j}$) $i, j = 1, 2, \dots, n$ рівний максимальному елементу в i -му рядку та j -му стовпчику таблиці A

Технічні умови. Програма **Table2021** читає з пристрою стандартного введення в першому рядку число n , а далі n рядків по n чисел в кожному – елементи таблиці A. Числа в рядках відокремлені пропусками. Програма виводить на пристрій стандартного введення – n рядків та n стовбців масиву B. Числа в рядках розділені пропусками.

Приклад

3	2	1	5
7	6	3	2
1	0	2	5
2	0	1	3

Введення

4

3 2 1 5

7 6 3 2

1 0 2 5

2 0 1 3

Виведення

7 6 5 5

7 7 7 7

7 6 5 5

7 6 3 5

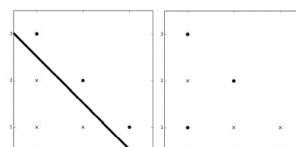
Задача Time2021. При авіаційному сполученні використовують місцевий час аеропортів вильоту та прибуття. Часові пояси характеризуються різницею в часі з меридіаном, на якому знаходиться Грінвічська обсерваторія. Для кожного часового поясу вводиться відхилення від UTC (Всесвітній координований час). Наприклад, місто аеропорт A знаходиться в часовому поясі UTC+3, а аеропорт B - UTC+7. Якщо рейс з A в B триває рівно 4 години, літак з A вилетів о 11:15, то в B він прибуде о 19:15 (4 годин польоту+4 години різниця в часі). Часові пояси змінюються від UTC-11 до UTC+14. По заданому часу вильоту та часу польоту, а також по годинним поясам аеропортів вильоту та прибуття треба визначити місцевий час прильоту та кількість днів, проведених в польоті.

Технічні умови. Програма **Time2021** читає з пристрою стандартного введення в першому рядку 2 числа через пропуск H_D, M_D ($0 \leq H_D \leq 23, 0 \leq M_D \leq 59$) – час вильоту, в другому рядку H_F, M_F ($0 \leq H_F \leq 10^9, 0 \leq M_F \leq 59$) – час польоту, в третьому рядку 2 цілих числа D, A ($-11 \leq D, A \leq 14$) - часові пояси аеропортів вильоту та прильоту. Програма виводить на пристрій стандартного виведення три числа в одному рядку через пропуски $H_A, M_A, Days$ час польоту в годинах та хвилинах, а різницю в датах вильоту та прильоту.

Приклади

Введення	Виведення	Коментар
11 15 4 0 3 7	19 15 0	Відповідає прикладу в задачі
12 0 1 0 -10 13	14 0 1	В цьому прикладі літак перетнув лінію зміни дати

Задача learning. У машинному навчанні часто виникає завдання лінійної класифікації об'єктів, коли класи об'єктів поділяються між собою лінійною поверхнею. Наприклад, у нас є інформація про кількість днів з моменту реєстрації облікового запису в соціальній мережі та кількість надісланих повідомлень за останній день, а також інформація про те, чи є цей обліковий запис спам-ботом. Вік облікового запису ми можемо взяти за X координату точки, а кількість повідомлень за Y координату. Завдання класифікації полягає в тому, щоб провести якусь пряму так, щоб об'єкти одного типу знаходилися але одну сторону цієї прямої, а об'єкти іншого типу по іншу.



За наявності такої прямої ми зможемо прогнозувати тип навіть незнайомого об'єкта при відомому віку облікового запису та кількості відправлених повідомлень залежно від того, з якого боку від прямої виявився об'єкт. Звичайно, у реальних даних можуть бути помилки вимірів або незвичайні об'єкти і провести таку пряму не завжди можливо, тому що, наприклад, об'єкт першого типу може випадково потрапити в скупчення об'єктів другого типу і відокремити його прямо неможливо.

Вам необхідно за інформацією про параметри та тип об'єктів визначити, чи існує пряма, яка однозначно поділяє класи об'єктів. Пряма не повинна проходити ні через один об'єкт.

Технічні умови. Програма Learning читає з пристрою стандартного введення кілька тестових блоків. У першому рядку задано число T - кількість тестових блоків ($1 \leq T \leq 100$). Кожен тестовий блок складається з числа N - кількість описаних об'єктів ($1 \leq N \leq 2000$). У наступних N рядок міститься опис об'єктів, що складаються з трьох цілих чисел X, Y, Type ($0 < X, Y < 10^7$, $0 < \text{Type} < 1$). Програма виводить на пристрій стандартного виведення для кожного

тестового блоку YES, якщо поділ можливий і NO, якщо ні, для кожного блоку в окремій стрічці.

Пример

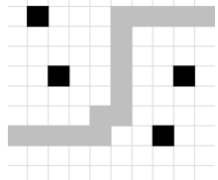
Введення	Виведення
2	YES
6	NO
1 1 1	
1 2 1	
1 3 0	
2 1 1	
2 2 0	
3 1 0	
6	
1 3 0	
2 2 0	
1 2 1	
3 1 1	
2 1 1	
1 1 0	

Задача Toxins. У популярній настільній грі завдання полягає в тому, щоб провести фішку по певному маршруту. Поле є прямокутником, що складається з квадратних клітинок. Гравець може за один хід перейти в одну з чотирьох сусідніх по стороні клітинок, не виходячи при цьому за межі поля. Гравець починає свій шлях у будь-якій клітинці самого лівого стовпця і повинен потрапити до будь-якої клітинки самого правого стовпця ігрового поля.

Але окремі клітинки токсичні, їх розташування відоме. Вася з'ясував, що чим більша відстань від токсичної клітинки, тим безпечніший маршрут, відстань обраховується як кількість ходів клітинками токсичних клітинок до кожної клітинки маршруту. Допоможіть йому визначити, на яку мінімальну відстань доведеться підійти до токсичної клітинки, рухаючись найбезпечнішим маршрутом.

Технічні умови. Програма **Toxins** читає з пристрою стандартного введення першому рядку записано натуральні числа N та M ($1 \leq N, M \leq 500$) — кількість рядків та стовпців на ігровому полі, у другому рядку записано натуральне число K ($1 \leq K \leq 500$) — кількість токсичних клітинок. У наступних K рядок записані пари чисел $R_i C_i$ ($1 \leq R_i \leq N, 1 \leq C_i \leq M$) — координати токсичних клітинок (рядок, стовець). Програма виводить на пристрій стандартного виведення мінімальну відстань, на яку доведеться наблизитися до токсичної клітинки на безпечному маршруті.

Приклад

Введення	Виведення	Коментар
10 10 4 2 2 5 3 5 9 8 8	3	 <p>Приклад одного з безпечних маршрутів показано на малюнку. Токсичні клітинки – чорні, клітинки маршруту – сірі. Мінімальна відстань між токсичною клітинкою та маршрутом – 3 ходи</p>