# Лабораторна робота №4

# «Складені типи даних»

Варіант №1

1. Магічний квадрат - це квадратна таблиця n x n, заповнена цілими числами таким чином, що сума чисел у кожному рядку, кожному стовпчику і на обох діагоналях однакова. Напишіть програму, яка отримує послідовність чисел у форматі як у вхідних даних, з яких утворюється двовимірна таблиця (список списків) цілочисельних елементів і перевірте чи є вона магічним квадратом. Програма має вивести True чи False відповідно.

Вхідні дані:

1 2 3,4 5 6,7 8 9

4 9 2,3 5 7,8 1 6

Вихідні дані:

False

True

1. Дано ціле число n, за яким слідують n рядків тексту. Надрукуйте кількість різних слів, які з’являються у тексті. Для цього ми визначаємо, що слово - це послідовність символів, що не містить пропусків. Слова розділені одним або декількома пропусками або символами нового рядка. Знаки пунктуації є частиною слова в цьому визначенні.

Варіант №2

1. Анаграма - переставлення літер у слові, завдяки чому утворюється нове значення слова. Напишіть програму, яка перевіряє, чи є два введених слова анаграмами. Програма повинна вивести True в разі, якщо введені слова є анаграмами, і False в інших випадках. Слово може складатися тільки з англійських символів. Регістр букв не повинен впливати на відповідь.
2. Дано два списки чисел, які можуть містити до 10000 чисел кожен. Виведіть всі числа, які входять як в перший, так і в другий список в порядку зростання. Вводяться два списки цілих чисел. Всі числа кожного списку знаходяться на окремому рядку.

Варіант №3

1. У римській системі числення для позначення чисел використовуються наступні символи (праворуч записані числа, яким вони відповідають в десятковій системі числення):

* I = 1
* V = 5
* X = 10
* L = 50
* C = 100
* D = 500
* M = 1000

Будемо використовувати варіант, в якому числа 4, 9, 40, 90, 400 і 900 записуються як віднімання з більшого числа меншого: IV, IX, XL, XC, CD і CM, відповідно. Напишіть програму, яка переводить число з римської в десяткову систему числення. Користувач вводить рядок, що містить число, закодоване в римській системі числення. Гарантується, що число менше 4000. Програма має вивести рядок, що містить число в десятковій системі числення, відповідне введеному.

1. Напишіть програму, яка приймає на вхід список цілих чисел, розділених пропуском, і виводить на екран значення, які повторюються в ньому більш ніж один раз. При виведенні числа не повинні повторюватися, порядок виведення може бути довільним.

Варіант №4

1. Напишіть програму, яка знайде максимально можливе число, яке можна отримати після видалення однієї цифри. Перший рядок містить одне ціле число n (10 ≤ n ≤ 2025). Виведіть у другому рядку максимально можливе число, яке можна отримати після видалення однієї цифри.
2. Дано список країн і міст кожної країни. Потім подані назви міст. Для кожного міста вкажіть, в якій країні воно знаходиться. Програма отримує на вхід кількість країн n. Далі йде n рядків, кожен рядок починається з назви країни, потім йдуть назви міст цієї країни. У наступному рядку записано число m, далі йдуть m запитів - назви якихось m міст, перерахованих вище. Для кожного із запиту виведіть назву країни, в якій знаходиться дане місто.

Вхідні дані:

2

Ukraine Kyiv Kharkiv Lviv

England London Liverpool Manchester Bristol

3

Kyiv

London

Lviv

Вихідні дані:

Ukraine

England

Ukraine

Варіант №5

1. У трьох автобусах розсадили дітей так, що їх кількість в різних автобусах різна. Яку найменшу кількість дітей потрібно пересадити, щоб у кожному автобусі їх було порівно? У першому рядку задано три натуральних числа, не більших за 100 - кількість дітей у першому, другому та третьому автобусах. Виведіть одне число - найменшу кількість дітей, яких потрібно пересадити. Якщо це зробити не можливо, то виведіть NO SOLUTIONS.
2. Користувачем вводяться наступні дані: у першому рядку міститься кількість рядків n, а потім - вводяться n рядків слів. Надрукуйте слово, яке в тексті зустрічається найчастіше. Якщо таких слів є декілька, надрукуйте слово, яке в алфавітному порядку розташоване раніше.

Варіант №6

1. Вводиться послідовність цілих чисел у вигляді як у вхідних даних і перетворюється у список списків, тобто у список, кожен елемент якого також є списком. Впорядкуйте цей список по другому елементу кожного списку і виведіть впорядкований список.

Вхідні дані:

1 2 3,2 1 3,4 0 1

Вихідні дані:

[[1, 2, 3], [2, 1, 3], [4, 0, 1]]

[[4, 0, 1], [2, 1, 3], [1, 2, 3]]

1. На шаховій дошці стоїть кінь. Відзначте положення коня на дошці і всі клітинки, які б’є кінь. Клітинку, де стоїть кінь, відзначте буквою K, клітинки, які б’є кінь, відзначте символами \*, решта клітинок заповніть крапками. Програма отримує на вхід координати коня на шаховій дошці в шаховій нотації, тобто, у вигляді e2, де спочатку записується номер стовпця (буква від a до h, зліва направо), потім номер рядка (цифра від 1 до 8, знизу догори). Виведіть на екран зображення шахової дошки як у вихідних даних.

Вхідні дані:

e4

Вихідні дані:

. . . . . . . .

. . . . . . . .

. . . \* . \* . .

. . \* . . . \* .

. . . . K . . .

. . \* . . . \* .

. . . \* . \* . .

. . . . . . . .

Варіант №7

1. Напишіть програму, яка отримує на вхід чотиризначне додатне десяткове число та генерує з цифр цього числа мінімально можливе чотиризначне число та максимально можливе чотиризначне число. У першому рядку задано одне чотиризначне число. Виведіть у другому рядку два числа - найменше чотиризначне число та найбільше чотиризначне число, які можна отримати з цифр даного числа. Числа розділяйте одним пропуском.

Варіант №8

1. Кодування довжин послідовностей - це базовий алгоритм стиснення даних. Реалізуйте алгоритм дешифрування (декодування) рядків, закодованих за допомогою одного з найпростіших варіантів кодування довжин послідовностей. На вхід алгоритму подається рядок, що містить цифри і символи англійського алфавіту. Цей рядок розбивається на так звані серії (послідовності), які кодуються парою число-символ або просто символ (в такому випадку число вважається рівним одиниці). Результат повинен містити ці серії в тому ж порядку, що вони і зустрічаються в заданому рядку, при цьому кожна серія розгортається в послідовність символів відповідної довжини. Наприклад, закодований рядок 3ab4c2CaB розбивається на серії 3a b 4c 2C a B, після чого, перетворюючи серії, отримуємо початковий розкодований рядок aaabccccCCaB.
2. Дано текст в рядку. Виведіть слово, яке в цьому тексті зустрічається найчастіше. Якщо таких слів кілька, виведіть те, яке менше в лексикографічному порядку.

Варіант №9

1. Напишіть програму, яка знаходить усі позиції входження підрядка B у рядок A. На першому рядку міститься вхідний рядок, на другому рядку введення вказаний підрядок, позиції якого потрібно знайти. Рядки складаються з символів англійського алфавіту. Програма має вивести позиції входження підрядка B у рядок A (індексація починається з 1), розділені пропуском або число -1 в разі, коли підрядок не знайдено.
2. На шаховій дошці стоїть ферзь. Відзначте положення ферзя на дошці і всі клітинки, які б’є ферзь. Клітинку, де стоїть ферзь, відзначте буквою Q, клітинки, які б’є ферзь, відзначте символами \*, решта клітинок заповніть крапками. Програма отримує на вхід координати ферзя на шаховій дошці в шаховій нотації, тобто, у вигляді e2, де спочатку записується номер стовпця (буква від a до h, зліва направо), потім номер рядка (цифра від 1 до 8, знизу догори). Виведіть на екран зображення шахової дошки як у вихідних даних.

Вхідні дані:

f3

Вихідні дані:

\* . . . . \* . .

. \* . . . \* . .

. . \* . . \* . .

. . . \* . \* . \*

. . . . \* \* \* .

\* \* \* \* \* Q \* \*

. . . . \* \* \* .

. . . \* . \* . \*

Варіант №10

1. Напишіть програму, яка приймає на вхід список чисел і число (в окремих рядках), після чого виводить всі позиції через пропуск, на яких це число зустрічається в переданому списку (позиції у списку нумеруються з 1). Позиції повинні бути виведені в порядку зростання. Якщо число не знайдено в списку, потрібно вивести рядок None (без лапок, з великої літери).
2. На шаховій дошці стоїть слон. Відзначте положення слона на дошці і всі клітинки, які б’є слон. Клітинку, де стоїть слон, відзначте буквою B, клітинки, які б’є слон, відзначте символами \*, решта клітинок заповніть крапками. Програма отримує на вхід координати слона на шаховій дошці в шаховій нотації, тобто, у вигляді e2, де спочатку записується номер стовпця (буква від a до h, зліва направо), потім номер рядка (цифра від 1 до 8, знизу догори). Виведіть на екран зображення шахової дошки як у вихідних даних.

Вхідні дані:

g4

Вихідні дані:

. . \* . . . . .

. . . \* . . . .

. . . . \* . . .

. . . . . \* . \*

. . . . . . B .

. . . . . \* . \*

. . . . \* . . .

. . . \* . . . .

Варіант №11

1. Вводиться рядок, який може мати пропуски на початку, між словами і у кінці. Пропусків може бути більше одного. Необхідно привести рядок до нормального вигляду, тобто видалити усі пропуски, а між словами залишити тільки один пропуск.

Варіант №12

1. Дано список цілих чисел. Потрібно стиснути його, перемістивши всі ненульові елементи в ліву частину списку, не змінюючи їх порядок, а всі нулі - в праву частину. Порядок ненульових елементів змінювати не можна, додатковий список використовувати не можна, завдання потрібно виконати за один прохід по списку. Роздрукуйте отриманий список.
2. На шаховій дошці стоїть тура. Відзначте положення тури на дошці і всі клітинки, які б’є тура. Клітинку, де стоїть тура, відзначте буквою R, клітинки, які б’є тура, відзначте символами \*, решта клітинок заповніть крапками. Програма отримує на вхід координати тури на шаховій дошці в шаховій нотації, тобто, у вигляді e2, де спочатку записується номер стовпця (буква від a до h, зліва направо), потім номер рядка (цифра від 1 до 8, знизу догори). Виведіть на екран зображення шахової дошки як у вихідних даних.

Вхідні дані:

h5

Вихідні дані:

. . . . . . . \*

. . . . . . . \*

. . . . . . . \*

\* \* \* \* \* \* \* R

. . . . . . . \*

. . . . . . . \*

. . . . . . . \*

. . . . . . . \*

Варіант №13

1. один ряд поставили n кеглів, пронумерувавши їх зліва направо числами від 1 до n. Потім в цей ряд кинули k куль, при цьому i-та куля збила всі кеглі з номерами від m до h включно. Визначте, які кеглі залишилися стояти на місці. Програма отримує на вхід кількість кеглів n і кількість кидків k. Далі йде k пар чисел m, h, при цьому 1 ≤ m ≤ h ≤ n ≤ 100. Програма повинна вивести послідовність з n символів, де j-й символ є I, якщо j-та кегля залишилася стояти, або ., якщо j-та кегля була збита.

Вхідні дані:

10 3

8 10

2 5

3 6

Вихідні дані:

I.....I...

1. На шаховій дошці стоїть король. Відзначте положення короля на дошці і всі клітинки, які б’є король. Клітинку, де стоїть король, відзначте буквою K, клітинки, які б’є король, відзначте символами \*, решта клітинок заповніть крапками. Програма отримує на вхід координати короля на шаховій дошці в шаховій нотації, тобто, у вигляді e2, де спочатку записується номер стовпця (буква від a до h, зліва направо), потім номер рядка (цифра від 1 до 8, знизу догори). Виведіть на екран зображення шахової дошки як у вихідних даних.

Вхідні дані:

c5

Вихідні дані:

. . . . . . . .

. . . . . . . .

. \* \* \* . . . .

. \* K \* . . . .

. \* \* \* . . . .

. . . . . . . .

. . . . . . . .

. . . . . . . .

Варіант №14

1. Вводиться рядок слів, розділених пропусками. Знайти найдовше слово і вивести його на екран. Розглянути випадок, коли найдовших неоднакових слів може бути кілька.
2. Програма отримує на вхід координати певної клітинки на шаховій дошці в шаховій нотації, тобто, у вигляді e2, де спочатку записується номер стовпця (буква від a до h, зліва направо), потім номер рядка (цифра від 1 до 8, знизу догори). Виведіть на екран зображення шахової дошки, в якому вказаний колір клітинки: B (чорний колір) або W (білий колір).

Вхідні дані:

b6

Вихідні дані:

. . . . . . . .

. . . . . . . .

. B . . . . . .

. . . . . . . .

. . . . . . . .

. . . . . . . .

. . . . . . . .

. . . . . . . .

Варіант №15

1. Користувач вводить ціле число n і видаляє одну цифру числа так, щоб число що утворилось, було максимально можливим. Наприклад, якщо число, яке ввели 432, то необхідно видалити цифру 2, яка стоїть на 3 місці, щоб отримати максимально можливе число 43. Напишіть програму, яка знайде максимально можливе число, яке можна отримати після видалення однієї цифри.
2. Двоє друзів вирішили здійснити влітку сходження в гори Карпати. Кожен з них зібрав свій рюкзак речей. Перевірте, які речі присутні в обох рюкзаках друзів, які є лише в першому рюкзаку, але не має в другому і, навпаки, є в другому і відсутні в першому. Дано два словники, у яких ключами є назви речей, а значеннями - кількості речей. Надрукуйте назви речей і їх кількості, які присутні в обох рюкзаках друзів, лише в першому рюкзаку і лише в другому рюкзаку.

Варіант №16

1. Дано список з n (1 ≤ n ≤ 100000) цілих чисел і число k (|k| < 100000). Циклічно посуньте список на |k| елементів вправо, якщо k - додатне і вліво, якщо від’ємне число. Програма отримує на вхід список цілих чисел, потім число k. Рішення повинно мати складність O(n), тобто не повинно залежати від k. Додатковим списком користуватися не можна.

Варіант №17

1. Вводяться n рядків. Визначити найдовший рядок і вивести його номер на екран. Якщо найдовших рядків кілька, то вивести номери всіх таких рядків.
2. На вхід програми подається рядок, що містить слова, розділені пропуском. Програма повинна вивести статистику довжин слів в отриманому рядку, від меншої довжини слова до більшої. Словом вважається послідовність довільних символів, оточена пропусками або межами рядка. Зауважте, що розділові знаки також відносяться до слова. Користувач вводить один рядок, що містить послідовності англійських символів і знаків пунктуації, розділені пропуском. Для кожної довжини слова, що зустрічається в заданому рядку, потрібно вказати кількість слів з такою довжиною у форматі «довжина: кількість». Статистика повинна виводитися в порядку збільшення довжини.

Варіант №18

1. Вводиться рядок, що складається із слів, розділених пропусками. Потрібно впорядкувати слова в рядку так, щоб найбільш короткі з них були спочатку рядка, а найбільш довгі в кінці. Слова необхідно вивести на екран через кому і пропуск, а рядок впорядкованих слів має закінчуватися крапкою.
2. На вхід програми подається рядок, який містить одне ціле число n - кількість англійських слів у англійсько-латинському словнику, за якими йдуть n перекладів цих слів. Кожен запис міститься в окремому рядку, що містить перше англійське слово, потім дефіс, оточений пропусками, а потім розділений комами список з перекладами цього англійського слова на латинській мові. Всі слова складаються лише з англійських літер. Переклади сортуються в лексикографічному порядку. Порядок англійських слів у словнику також лексикографічний. Надрукуйте відповідний латинсько-англійський словник у тому ж форматі (кількість слів та їхні переклади). Зокрема, першим у рядку слів має бути лексикографічно мінімальний переклад латинського слова, потім другого в цьому порядку і т. д. Англійські слова в кожному рядку також слід сортувати лексикографічно.

Вхідні дані:

3

apple - malum, pomum, popula

fruit - baca, bacca, popum

punishment - malum, multa

Вихідні дані:

7

baca - fruit

bacca - fruit

malum - apple, punishment

multa - punishment

pomum - apple

popula - apple

popum – fruit

Варіант №19

1. Напишіть програму, на вхід якої подається список чисел одним рядком. Програма повинна для кожного елемента цього списку вивести суму двох його сусідів. Для елементів списку, які є крайніми, одним із сусідів вважається елемент, що знаходить на протилежному кінці цього списку. Якщо на вхід прийшло тільки одне число, треба вивести його ж. Виведення повинно містити один рядок з числами нового списку, розділеними пропусками.
2. Кожен з n школярів деякої школи знає m мов. Визначте, які мови знають всі школярі і мови, які знає хоча б один з школярів. Перший рядок вхідних даних містить кількість школярів n. Далі йде n чисел m, після кожного з чисел йде m рядків, що містять назви мов, які знає i-й школяр. Довжина назв мов не перевищує 1000 символів, кількість різних мов не більше 1000 (1 ≤ n ≤ 1000, 1 ≤ m ≤ 500). У першому рядку виведіть кількість мов, які знають усі школярі. Починаючи з другого рядка - список таких мов. Потім - кількість мов, які знає хоча б один школяр, на наступних рядках - список таких мов.

Варіант №20

1. Використовуючи поняття списку, напишіть програму, яка створює 3D масив елементів a x b x c, кожен з яких має значення 0. Значення a, b, c вводяться в одному рядку через пропуск.
2. Напишіть програму, яка для двох послідовностей, що складаються з натуральних чисел, що не перевершують n, буде визначати, які числа зустрічаються в кожній з послідовностей, а які з чисел від 1 до n - в жодній з них. Спочатку на вхід програмі подається число n (1 ≤ n ≤ 255). У другому рядку вхідних даних знаходяться елементи першої послідовності, у третьому рядку - елементи другої послідовності. Виведіть у першому рядку в порядку зростання без повторень числа, які зустрічаються в кожній з послідовностей, а у другому рядку - в порядку зростання числа від 1 до 255, які не зустрічаються в жодній з них.

Варіант №21

1. Суть завдання та ж, що і у задачі Шифр Цезаря, з однією відмінністю: кодуються символи з інтервалу 1F600-1F64F таблиці символів Unicode. Використовується кодування UTF-8. Для всіх символів зсув один і той же. Зсув циклічний, тобто, якщо до останнього символу алфавіту застосувати одиничний зсув, то він заміниться на перший символ, і навпаки. Напишіть програму, яка шифрує текст шифром Цезаря. На першому рядку вказується використовуваний зсув шифрування: ціле число. Додатне число відповідає зсуву вправо. На другому рядку вказується непорожня фраза для шифрування. Не звертайте уваги, якщо у вас відображаються прямокутники замість деяких символів. Це означає, що ваш шрифт не містить цих символів. Програма має вивести єдиний рядок, в якій записана зашифрована послідовність.
2. Вводиться число n, за яким слідують n рядків тексту. Напишіть програму, яка друкує всі слова, що зустрічаються в тексті, по одному на рядок, і їхню кількість входжень у текст. Слова повинні бути відсортовані в порядку спадання відповідно до їх кількості, і всі слова при однаковому числі входжень у текст повинні бути надруковані в лексикографічному порядку. Вказівка. Після того, як ви створите словник усіх слів, вам захочеться впорядкувати його по частоті слова. Бажаного можна домогтися, якщо створити список, елементами якого будуть кортежі з двох елементів: частота зустрічальності слова і саме слово. Наприклад, [(2, 'hi'), (1, 'what'), (3, 'is')]. Тоді стандартне сортування буде сортувати список кортежів, при цьому кортежі порівнюються по першому елементу, а якщо ці елементи рівні - то по другому.