

# МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних і контрольних робіт

по програмуванню мовою Паскаль

ч а с т и н а 1

(для студентів спеціальності 7.091501  
«Комп'ютерні мережі і системи»  
прискореної форми навчання)

Укладачі доц.Назаренко В.І.,  
ас.Демеш Н.С.  
ас.Юсупова К.Б.

Розглянуто  
на засіданні кафедри КІ  
протокол № 1 від 31 серпня 2010 р.

Затверджено  
на засіданні навчально-  
видавничої ради ДонНТУ  
протокол № 5 від 06.12.10

Донецьк ДонНТУ 2010

Навчальне видання

УДК 681.3(07)

Методичні вказівки до лабораторних і контрольних робіт по програмуванню мовою Паскаль, частина 1 (для студентів спеціальності 7.091501 “Комп'ютерні мережі і системи” прискореної форми навчання). Укладачі: В.І. Назаренко, Н.С.Демеш, .К.Б.Юсупова.- Донецьк: ДонНТУ, 2010. - 55 с.

Наведено методичні вказівки до лабораторної роботи № 1 і контрольної роботи № 1, що виконуються у першому семестрі відповідно до навчального плану спеціальності «Комп'ютерні мережі і системи» прискореної форми навчання. По лабораторній роботі № 1 «Обробка одновимірних масивів» наведено 50 варіантів завдань, стільки ж завдань записано для контрольної роботи № 1 «Обробка матриць». Матеріал посібника містить також загальні методичні вказівки, спрямовані на формування у студентів сучасного стилю програмування.

Укладачі: доц. Назаренко В.І.,  
ас. Демеш Н.С.  
ас. Юсупова К.Б.

Відповідальний  
за випуск проф. Святний В.А.

Рецензент доц. Федяєв О.І.

## ЗАГАЛЬНІ МЕТОДИЧНІ ВКАЗІВКИ

Цикл лабораторних і контрольних робіт із програмування мовою Турбо Паскаль призначений для придбання практичних навичок алгоритмізації інженерно-технічних завдань та їхньої програмної реалізації. Послідовність виконання завдань побудована за принципом зростаючої складності й поступового використання в програмах мовних засобів, які розглядаються в лекційному курсі програмування. Конкретні рекомендації з виконання контрольних робіт наведені в методичних вказівках до кожної роботи.

У загальних методичних вказівках розглядаються основні вимоги, пов'язані з формуванням у студентів правильного стилю програмування.

Основним принципом у проектуванні програм є наступний: програма складається насамперед для людини, а не для машини. У зв'язку із цим у тексті програми обов'язково повинні утримуватися засоби, які полегшують її розуміння й поліпшують її читабельність.

Спеціальних стандартів на тексти програм не існує. Проте професійні програмісти при написанні тексту програми дотримуються певних правил, спрямованих на додержання наведеного вище принципу програмування.

При розробці програми, що реалізує завдання по контрольній роботі, обов'язковим стосовно тексту програми є виконання наведених нижче основних вимог.

1. Заголовки програми й кожного розділу (**Program**, **Label**, **Const**, **Type**, **Var**, **Begin** розділу операторів), а також фразу **Uses** і заголовки процедур і функцій починати з першої позиції рядка.

2. Розділи описів, як правило, розташовувати в порядку, передбаченому стандартною мовою Паскаль: **Label**, **Const**, **Type**, **Var**, опис процедур і функцій.

3. Кожне слово **Begin** починати з нового рядка.

4. Кожне слово **End** записувати з нового рядка саме з тієї позиції, з якої починається стосовне до нього слово: **Begin**, **Case** або **Record**.

5. Текст програми між словами **Begin** і **End**, **Case** і **End**, **Record** і **End** повинен бути зрушений на один відступ праворуч. Відступ дорівнює двом позиціям рядка.

6. Текст програми після слів **do**, **then**, **else** повинен починатися з нового рядка й бути зрушений на один відступ праворуч стосовно початку попереднього рядка.

7. Кожне слово **Else** повинне бути розташоване під тим словом **If**, до якого воно ставиться.

8. Після заголовка процедури або функції приводити в коментарі короткий опис роботи, що вона виконує.

9. Довгі тексти програми розділяти коментарями на фрагменти. У коментарі стисло описувати роботу фрагмента.

10. Процедури й функції відокремлювати одну від іншою пунктирною лінією, узятій в коментар.

11. Після слова **End** завершального блоку процедури або функції в коментарі вказувати ім'я даної процедури або функції.

12. Після оголошення імені змінної, як правило, описувати в коментарі її призначення.

13. Як символи коментарю використовувати тільки фігурні дужки.

14. Імена елементів програми (змінні, масиви й ін.) записувати, як правило, із великої букви.

15. Імена змінних у програмі в загальному випадку не повинні відрізнятися від позначень, прийнятих у постановці завдання.

16. Окремі частини імені, що складається з декількох слів, виділяти великими буквами (наприклад, `FromStringToNumber`).

17. Довжина рядка програми не повинна перевищувати 68 символів (для друку тексту програми стандартним шрифтом на папері формату А4).

18. Оператор **Goto** дозволяється застосовувати в програмі у двох випадках:

- для примусового виходу із циклу (можна замінити оператором **Break**);

- для переходу до віддаленого фрагмента програми.

Абсолютно неприпустимо використовувати оператор **Goto** для переходу знизу нагору в програмі.

Якщо за допомогою оператора **Goto** відбувається перехід на кінець процедури або функції, то його доцільно замінити оператором **Exit**.

19. У програмі замість значень констант використовувати, як правило, імена констант.

20. У розділі опису змінних, як правило, використовувати імена типів, а не описи типів. Наприклад, замість опису

```
Var A : array[1..100] of string[40];
```

доцільно використати таку конструкцію:

```
Const Nmax = 100;
```

```

Type  string40 = string[40];
        Ar = array[1..Nmax] of string40;
Var    A : Ar;

```

Остання вимога зв'язана в основному з тим, що у великій програмі будь-яка змінна може бути використана надалі як фактичний параметр при звертанні до процедури, а при такому звертанні завжди потрібен збіг імені типу формального й фактичного параметрів.

21. Варто уникати різних програмних "трюків", що утрудняють розуміння програми.

Наприклад, важко зрозуміти роботу такого фрагмента:

```

Var  x, y, z, w : real;
.....
If  x>y then
    w:=0
Else
    w:=y-x;
    z:=x+w;

```

Виявляється, це обчислення значення, рівного більшому із двох чисел: x та y.

Другий приклад.

Для обміну значень двох змінних іноді записують:

```

x:=x-y;  y:=x+y;  x:=y-x;

```

замість більш ясної схеми обміну "по трикутнику":

```

z:=x;  x:=y;  y:=z .

```

Економія однієї змінної, що має місце в першому варіанті обміну, не виправдує зниження читабельності програми. До того ж перший варіант вимагає більше машинного часу для свого виконання.

Програмування відповідно до наведених вище правил часто іменується структурним програмуванням, а написану в такому стилі програму називають структурованою.

У методичних вказівках до кожної лабораторної і контрольної роботи наведені приклади її реалізації; програмні тексти прикладів наочно ілюструють виконання перерахованих вище вимог.

Тексти методичних вказівок розміщуються не тільки у виданих друкованим способом посібниках, вони записані також на комп'ютерах дисплейного класу. Студентам рекомендується переписати тексти методичних вказівок на дискету й використати надалі програмні приклади, які втримуються в них, як основу для розробки свого варіанта виконання лабораторної або

контрольної роботи. Насамперед ця рекомендація ставиться до сервісних процедур і функцій. Включення їх у свою програму шляхом копіювання файлу принаймні заощаджує час, що нерационально витрачається при наборі цих процедур із клавіатури.

## **Лабораторна робота № 1**

### **ОБРОБКА ОДНОВИМІРНИХ МАСИВІВ**

#### Методичні вказівки

Мета роботи - овоїти методи організації програм обробки одновимірних масивів.

У кожному варіанті завдань до лабораторної роботи №1 необхідно сформулювати один або два одновимірних масиви по наведеним у завданні формулам. Вхідні дані студент вибирає самостійно, але так, щоб для кожної логічної галузі програми були зазначені дані, які перебувають у необхідному діапазоні. Кількість елементів вхідного масиву рекомендується вибирати в межах 10 .. 20.

У програмі повинен бути передбачено увід і друк вхідних даних, а також друк вихідних результатів. Для контролю правильності обчислень може бути передбачено також друк проміжних результатів.

Вхідні дані в лабораторній роботі № 1 - це масив, що містить 10 .. 20 елементів. Оскільки при відладці програми користувач змушений багаторазово виконувати її запуск, то при кожному запуску програми було б потрібно заново виконувати набір із клавіатури всього масиву вхідних даних. Це помітно збільшує загальний час відладки програми і, крім того, пов'язане з потенційною можливістю внесення помилок у програму при формуванні вхідних даних вручну.

У роботі №1 вхідний масив даних повинен вводитися з текстового файлу. Текстовий файл формується в середовищі редактора Турбо Паскаля в такий же спосіб, як і Паскаль-програма (програма на Паскалі - це також текстовий файл). Оскільки при кожному запуску програми вхідні дані вводяться з того самого файлу, то використання файлів з вхідними даними виключає можливість внесення в програму помилок уведення, а також зменшує витрати часу програміста на відладку програми.

Кожен оброблюваний у Паскаль-програмі об'єкт повинен мати власне ім'я, що описується в розділі **Var**. Отже, у цьому розділі повинне бути зазначене також ім'я текстового файлу, з якого треба виконувати уведення вхідних даних. Текстовому файлу в розділі **Var** ставиться у відповідність стандартне ім'я типу `text`.

На відміну від інших об'єктів програми стосовно файлу розрізняють внутрішнє й зовнішнє імена. Внутрішнє ім'я – це ім'я, що привласнене файлу в розділі **Var**; зовнішнє - це ім'я, що має файл на зовнішньому носії інформації (у частковому випадку на твердому або на гнучкому диску). Внутрішнє ім'я формується за правилами мови Паскаль, зовнішнє - за правилами операційної системи MS DOS. У принципі тому самому файлу в програмі можуть бути поставлені у відповідність різні файли на диску.

Установка відповідності між зовнішнім і внутрішнім іменами файлу здійснюється процедурою `Assign`, що записується у вигляді

$$\text{Assign}(F, S),$$

де  $F$  - внутрішнє ім'я файлу;

$S$  - зовнішнє ім'я файлу, що записується в найпростішому випадку у вигляді рядка-константи.

Наприклад,

$$\text{Assign}(F, 'InArray.dat').$$

Файл у програмі повинен бути відкритий для його використання, а після закінчення обробки повинен бути закритий. Відкриття вхідного текстового файлу здійснюється процедурою

$$\text{Reset}(F),$$

його закриття - процедурою

$$\text{Close}(F).$$

Основна робота, що виконується операційною системою при відкритті файлу - це призначення спеціальної області оперативної пам'яті для організації вводу-виводу (буфер вводу-виводу); при закритті файлу ця область звільняється.

Уведення з текстового файлу й вивід у текстовий файл виконуються в такий же спосіб, як і введення із клавіатури й вивід на екран, але в процедурах `Read` і `Write` перед списком вводу-виводу повинне бути зазначене внутрішнє ім'я файлу.

При виконанні завдання по лабораторній роботі №1 пропонується дотримуватися наведених нижче вказівок.

### 1. Друк вхідних даних.

Всі вхідні дані, які вводяться ззовні (клавіатура, диск), повинні бути надруковані для візуального контролю. Можливий варіант програми виводу одновимірного масиву на екран наведений у прикладі виконання завдання по лабораторній роботі №1.

### 2. Програмування полінома.

Якщо формула, що програмується, - це поліном, то його варто обчислювати за схемою Горнера. У загальному випадку для полінома

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

схема Горнера має вигляд

$$P_n(x) = (\dots((a_n x + a_{n-1})x + \dots + a_1)x + a_0$$

Наприклад, вираз

$$y = 8x^5 - 2x^4 + x^2 - 10x + 1$$

за схемою Горнера повинен бути представлений у вигляді

$$y = (((8x - 2)x \cdot x + 1)x - 10)x + 1$$

Як відомо, погрішність добутку дорівнює сумі погрішностей співмножників. Оскільки для одержання значення полінома по першій формулі потрібно  $2n$  множень, а за схемою Горнера –  $n$  множень, то схема Горнера дозволяє одержати значення полінома з меншою погрішністю; а розрахунок за схемою Горнера також вимагає менше машинного часу. Додатковою перевагою схеми Горнера є можливість легко організувати обчислення полінома в циклі, якщо представити його коефіцієнти у вигляді одновимірного масиву.

*Приклад.* Скласти програму обчислення значення полінома  $n$ -ого ступеня.

Запишемо поліном  $n$ -ого ступеня в наступному вигляді:

$$P(x, n) = (\dots(a_n x + a_{n-1})x + \dots + a_1)x + a_0 .$$

Програма обчислення значення полінома:

```

Program Polynom;
Const Nmax = 100;
Type PolynomType = array[0..Nmax] of real;
Var
    i,                { параметр циклу }
    n : byte;        { ступінь полінома }
    x,                { аргумент }
    P : real;        { значення полінома }
    A : PolynomType; { коефіцієнти полінома }
Begin
    Уведення і друк змінної n і масиву A
    P:=0;
    For i:=n downto 0 do
        P:=P*x+a[i];
    Друк значення P
End.

```



### 3. Програмування степінної функції.

У Турбо Паскалі відсутня операція зведення в степінь, тому її реалізація виконується програмним методом.

Якщо показник степіня  $n$  - ціле додатне число, то операція зведення в степінь може бути замінена багаторазовим множенням (для цілочисельного показника степіня на Паскалі розроблені ефективні програми зведення в степінь, але ці програми мають порівняно складну структуру алгоритму їх організації). Якщо показник степіня - дійсне число, то обчислення функції

$$y = a^b$$

виконується по формулі

$$y = e^{b \ln a}$$

Тут варто звернути увагу на випадок, коли функцією, що обчислюється, є корінь непарного ступеня. Оскільки значення такої функції існує і при додатному, і при від'ємному аргументі, а функція  $\ln(x)$  визначена тільки для додатного аргументу, то обчислення кореня непарного ступеня

$$y = \sqrt[n]{x}$$

де  $n$  - ціле непарне число, можна організувати, наприклад, у такий спосіб:

```
If abs (x) < 1E-10 then  
  y := 0  
Else  
  Begin  
    y := exp (ln (abs (x)) / n) ;  
    If x < 0 then  
      y := -y ;  
  End;
```

*Примітка.* Записану вище програму обчислення кореня можна використати для будь-якого цілого додатного показника ступеня. При цьому для парного показника гарантується значення  $y=0$ , якщо аргумент  $x$  досить мале число.

### 4. Порівняння дійсних чисел.

Цілочисельні змінні визначають у програмі, як правило, об'єкти, які по своєму фізичному змісту можуть мати тільки цілі значення (наприклад, кількість елементів у масиві, порядковий номер елемента і т.п.). Значення таких змінних мають точне подання на машинному рівні, операції відносини для них завжди виконуються коректно.

Дійсні змінні, як правило, відображають у програмі параметри вимірюваних об'єктів (довжина, площа, маса, інтервал часу і т.п.). Оскільки виміри завжди виконуються з певною погрешністю, то значення дійсних змінних у загальному випадку вважаються наближеними.

Машинне зображення дійсних змінних також є наближеним, його точність визначається кількістю розрядів у мантисі, що в більшості випадків веде до округлення значення такої змінної. Наприклад, десяткове число 0.9 у двійковій системі числення – це нескінченний дріб  $0.1111001100110011\dots = 0.11(1100)$ . У цьому випадку машинне зображення числа у форматі real дорівнює вихідному значенню лише з точністю останнього розряду мантиси.

Безпосереднє порівняння дійсних змінних у більшості випадків є некоректним. Припустимо для приклада, що маси  $P_1$  і  $P_2$  двох вантажів обмірювані з точністю 0.1 кг. Якщо ці маси рівні, то в програмі потрібно виконати оператор  $S_1$ , у протилежному випадку – оператор  $S_2$ . Тоді маємо:

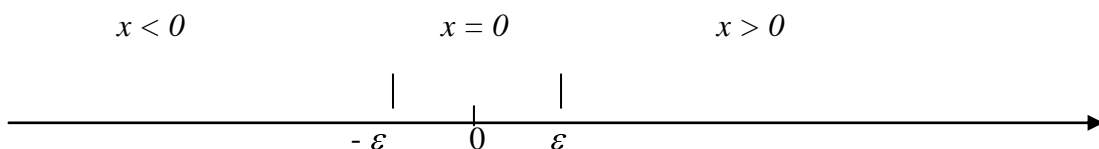
```

Var P1,P2: real;
.....
if P1=P2 then S1 else S2;

```

Якщо  $P_1 = 110.301$ , а  $P_2 = 110.302$ , то результат роботи програми буде невірним. Очевидно, у цьому випадку  $P_1$  і  $P_2$  потрібно вважати рівними, якщо вони відрізняються між собою не більше ніж на 0.1.

У загальному випадку порівняння дійсних змінних варто виконувати стосовно малого числа  $\varepsilon$ , що відображає погрішність подання таких змінних. Зокрема, порівняння з нульовим значенням наочно ілюструється на числовій осі:



Отже,  $x = 0$ , якщо  $abs(x) \leq \varepsilon$ ;  $x > 0$ , якщо  $x > \varepsilon$ ;  $x < 0$ , якщо  $x < -\varepsilon$ .

Аналогічно замість порівняння  $a > b$  варто аналізувати відношення  $a - b > 0$ .

### **Звіт по лабораторній роботі №1**

У звіті по лабораторній роботі необхідно привести:

- титульний аркуш;
- номер варіанта й умова завдання;
- короткий опис програми;
- контрольний розрахунок;
- текст програми;

- результати роботи програми.

Матеріал лабораторної роботи повинен бути представлений на аркушах формату А4. При цьому на титульному аркуші повинне бути написано:

## ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Кафедра комп'ютерної інженерії

Лабораторна робота № 1  
за курсом «Програмування»

ст-та групи (шифр групи)

Прізвище й ініціали

Номер залікової книжки

Домашня адреса:

Текст програми (листинг) і результати рішення завдання повинні бути надруковані на принтері.

Рекомендується текст програми друкувати за допомогою компілятора мови Турбо Паскаль. Якщо ж друк виконується в середовищі текстового процесора Word, то в цьому випадку потрібно використати шрифт Courier New (цей шрифт завжди відводить одну позицію для кожного символу, що забезпечує наочність структурованої програми). Для описів умови завдання, алгоритму й програми доцільно застосовувати шрифт Times New Roman.

Для контрольного розрахунку, виконуваного за допомогою калькулятора, потрібно брати 2-3 значення вхідних даних таким чином, щоб для кожної галузі в розгалуженні було задано хоча б одне число. При цьому потрібно приводити не кінцевий результат обчислень, а поетапний, для окремих частин використовуваних формул. Робота програми буде вважатися коректною, якщо її результати збігаються з контрольним розрахунком.

У методичних вказівках наведено 70 варіантів завдань. Вибір варіанта виконується за узгодженням з викладачем, що веде в навчальній групі лабораторні або практичні заняття по програмуванню.

### Приклад виконання завдання

$$y_i = 2 \sin^2(\pi x_i) - \cos(x_i^2)$$
$$z_i = \begin{cases} x_i^3 + 2x_i^2 + x_i - 1 & , \text{ якщо } y_i > 1 \\ 1 - x_i^2 & , \text{ якщо } y_i \leq 1 \end{cases}$$

Потрібно знайти довжини векторів  $Y$  і  $Z$ :

$$L_y = \sqrt{\sum_{i=1}^n y_i^2}; \quad L_z = \sqrt{\sum_{i=1}^n z_i^2},$$

а також визначити, скільки пар елементів  $(y_i, z_i)$  збігаються за знаком.

У цьому завданні вхідними даними є елементи одновимірному масиву (вектора)  $X$ , які використовуються для обчислення елементів  $y_i$  і  $z_i$ . Кінцевим результатом роботи програми є довжини векторів  $L_y$  й  $L_z$ , а також кількість  $m$  співпадаючих за знаком елементів.

*Контрольний розрахунок.*

Розділимо обчислення значення  $y_i$  на дві частини:

$$R1 = \sin^2(\pi x_i) \text{ и } R2 = \cos(x_i^2)$$

для  $x = 1.5$  будемо мати  $R1 = 2.00$ ;  $R2 = -0.63$ ;  $y = 2.63$ ;  $z = 8.38$ .

Аналогічно для  $x = 6.6$  одержимо  $R1 = 1.89$ ;  $R2 = 0.91$ ;  $y = 0.90$ ;  $z = -42.56$ .

Програмна реалізація розглянутого приклада виконана у двох варіантах: без використання процедур і функцій (Cont1a) і з їхнім використанням (Cont1b). Наявність цих двох варіантів покликано продемонструвати переваги застосування процедур і функцій для подання однотипних фрагментів програми. Студентам рекомендується виконувати отримані завдання по другому варіанті, тобто з використанням процедур і функцій.

```
Program Cont1a;  
Uses Crt;  
Const Nmax = 100;  
      eps = 0.001;  
Type Ar = array[1..Nmax] of real;  
Var  
  Sy, Sz : shortint; { знаки елементів y[i] і z[i] }  
  i,      { параметр циклу }  
  n,      { кількість елементів масиву }  
  m,      { кількість співпадаючих }  
          { за знаком елементів }  
  k : byte; { допоміжна змінна }  
  Ly, Lz : real; { довжини векторів }
```

```

        ch : char;          { СИМВОЛ НАТИСНУТОЇ КЛАВІШІ }
        X,Y,Z : Ar;        { МАСИВИ }
        F : text;          { ВХІДНИЙ ФАЙЛ }
Begin
{ Уведення вхідних даних }
ClrScr;
Assign(F, 'X.dat');
Reset(F);
n:=0;
While not SeekEof(F) do
    Begin
        Inc(n);
        Read(F,x[n]);
    End;
Close(F);

{ Вивід вхідного масиву X на екран }
Writeln('Масив X   n=',n);
k:=0;
For i:=1 to n do
    Begin
        k:=k+1;
        If k<5 then
            Write(x[i]:8:2,' ')
        Else
            Begin
                k:=0;
                Writeln(x[i]:8:2);
            End
    End;
If k>0 then
    Writeln;

{ Обчислення елементів масивів Y і Z }
For i:=1 to n do
    Begin
        y[i]:=2*sqr(sin(pi*x[i]))-cos(sqr(x[i]));
        If (y[i]-1)>eps then
            z[i]:=((x[i]+2)*x[i]+1)*x[i]-1
        Else
            z[i]:=1-sqr(x[i]);
    End;

{ Вивід елементів масиву Y на екран }
Writeln('Масив Y');
k:=0;
For i:=1 to n do
    Begin
        k:=k+1;
        If k<5 then
            Write(y[i]:8:2,' ')
        Else
            Begin
                k:=0;
                Writeln(y[i]:8:2);
            End
    End;

```

```

        k:=0;
        Writeln(y[i]:8:2);
    End
End;
If k>0 then
    Writeln;

{ Вивід елементів масиву Z на екран }
Writeln('Масив Z');
k:=0;
For i:=1 to n do
    Begin
        k:=k+1;
        If k<5 then
            Write (z[i]:8:2,' ')
        Else
            Begin
                k:=0;
                Writeln(z[i]:8:2);
            End
    End;
If k>0 then
    Writeln;

{ Визначення довжин векторів Y і Z }
Ly:=0;
For i:=1 to n do
    Ly:=Ly+sqr(y[i]);
Ly:=sqrt(Ly);
Writeln('Ly=',Ly:8:2);
Lz:=0;
For i:=1 to n do
    Lz:=Lz+sqr(z[i]);
Lz:=sqrt(Lz);
Writeln('Lz=',Lz:8:2);

{ Визначення кількості співпадаючих за знаком }
{ елементів Y і Z }
m:=0;
For i:=1 to n do
    Begin
        If abs(y[i])<=eps then
            Sy:=0
        Else
            If y[i]>eps then
                Sy:=1
            Else
                Sy:=-1;
        If abs(z[i])<=eps then
            Sz:=0
        Else
            If z[i]>eps then
                Sz:=1
            Else

```

```

        Sz:=-1;
    If Sy=Sz then
        Inc(m);
    End;
    Writeln('Кількість співпадаючих за знаком ел-тів'+
        ' m = ',m);

{ Припинення роботи програми }
Repeat
    ch:=ReadKey;
Until ord(ch)=27;

End.

```

Результати роботи програми:

```

Масив X           n=10
    1.50    6.60    8.00    1.00    0.80
   -1.00    0.00   12.20   10.00  -10.00

```

```

Масив Y
    2.63    0.90   -0.39   -0.54   -0.11
   -0.54   -1.00    1.07   -0.86   -0.86

```

```

Масив Z
    8.38   -42.56   -63.00    0.00    0.36
    0.00    1.00  2124.73  -99.00  -99.00

```

```

Ly=      3.48    Lz=  2130.71
Кількість співпадаючих за знаком ел-тів m = 5

```

### *Коментарі до програми*

1. Фраза **Uses Crt** означає, що в програмі використовуються процедури й функції, які записані в стандартному модулі **Crt**. У даному випадку це відноситься до процедури **ClrScr** (очищення екрана) і функції **ReadKey** (читання символу натиснутої клавіші) .

2. Функція знака приймає одне із трьох значень: 0, 1, -1.

У зв'язку із цим для її зображення в програмі використовується тип короткого цілого зі знаком (**shortint**).

3. Уведення вхідних даних організовано по кінцю файлу. Цикл

```
While not SeekEof(F) do ...
```

працює, поки не будуть прочитані всі числа, що записані у файлі **F**. Одночасно в цьому циклі формується кількість уведених елементів **n**.

*Примітка.* Функція `SeekEof(F)` приймає значення `true`, якщо досягнуто маркер кінця файлу. Отже, `not SeekEof(F) = true`, якщо файл `F` ще не вичерпаний.

4. Константа  $\pi$  (`pi`) у Турбо Паскалі вважається визначеною, тому в програмі `Cont1` значення цієї константи не вказується.

5. Процедура `Inc(m)` еквівалентна операторові `m:=m+1`, але працює приблизно на 30 % швидше. Крім інкремента `Inc(m)`, у Паскалі є також декремент `Dec(m)`, що еквівалентний у цьому випадку операторові `m:=m-1`.

6. Оператор `ord(ch)` визначає порядковий номер символу `ch` у таблиці ASCII; 27 - це код клавіші `Esc`. Отже, наведений вище фрагмент програми припиняє її роботу тоді, коли буде натиснута клавіша `Esc`.

7. Програми виводу масивів `X`, `Y` і `Z` відрізняються між собою лише ім'ям масиву. Їх доцільно оформити у вигляді процедури. При цьому скорочується текст програми й поліпшується її читабельність. Таке ж зауваження можна зробити із приводу обчислення параметрів  $S_y$  і  $S_z$ ,  $L_y$  і  $L_z$ .

8. Нижче наведений текст програми `Cont1b` з використанням процедур і функцій. Уведення вхідного масиву й припинення роботи програми тут також оформлені у вигляді процедур, що дозволяє використати їх практично без змін в інших програмах.

```
Program Cont1b;
Uses Crt;
Const Nmax = 100;
        eps = 0.001;
Type Ar = array[1..Nmax] of real;
Var
    Sy,Sz : shortint;  { знаки елементів y[i] і z[i] }
        i,             { параметр циклу }
        n,             { кількість елементів масиву }
        m,             { кількість співпадаючих }
                       { за знаком елементів }
    Ly,Lz : real;      { довжини векторів }
    X,Y,Z : Ar;        { масиви }
        F : text;      { вихідний файл }
{ ----- }
Procedure WaitEscape;
{ Припинення програми при натисканні клавіші Esc }
Var ch: char;
Begin
    Repeat
        ch:=ReadKey;
    Until ord(ch)=27;
End { WaitEscape };
{ ----- }
```



```

Procedure ReadArray(Var X:Ar; Var n:byte);
{ Уведення одновимірного масиву }
Begin
  Reset(F);
  n:=0;
  While not SeekEof(F) do
    Begin
      Inc(n);
      Read(F,x[n]);
    End;
  Close (F);
End { ReadArray };
{ ----- }
Procedure WriteArray(S:string; Var X:Ar; n:byte);
{ Вивід на екран одновимірного масиву }
Var i,k: byte;
Begin
  Writeln(S, 'n=',n);
  k:=0;
  For i:=1 to n do
    Begin
      k:=k+1;
      If k<5 then
        Write(x[i]:8:2, ' ')
      Else
        Begin
          k:=0;
          Writeln(x[i]:8:2);
        End
    End;
  If k>0 then
    Writeln;
End { WriteArray };
{ ----- }
Function Sign(x:real):shortint;
{ Обчислення функції знака }
Begin
  If abs(x)<=eps then
    Sign:=0
  Else
    If x>eps then
      Sign:=1
    Else
      Sign:=-1;
End { Sign };
{ ----- }
Function Len(Var U:Ar; n:byte):real;
{ Обчислення довжини вектора U }
Var i : byte;
  L : real;
Begin
  L:=0;
  For i:=1 to n do
    L:=L+sqr(u[i]);

```

```

    L:=sqrt(L);
    Len:=L;
End { Len };
{ ----- }
Begin

{ Уведення й вивід вхідних даних }
ClrScr;
Assign(F, 'X.dat');
ReadArray(X,n);
WriteArray('Масив X',X,n);

{ Обчислення елементів масивів Y і Z }
For i:=1 to n do
    Begin
        y[i]:=2*sqr(sin(pi*x[i]))-cos(sqr(x[i]));
        If (y[i]-1)>eps then
            z[i]:=((x[i]+2)*x[i]+1)*x[i]-1
        Else
            z[i]:=1-sqr(x[i]);
        End;

{ Вивід на екран елементів масивів Y і Z }
WriteArray('Масив Y',Y,n);
WriteArray('Масив Z',Z,n);

{ Визначення довжин векторів Y і Z }
Ly:=Len(Y,n);
Writeln('Ly=',Ly:8:2);
Lz:=Len(Z,n);
Writeln('Lz=',Lz:8:2);

{ Визначення кількості співпадаючих за знаком }
{ елементів Y і Z }
m:=0;
For i:=1 to n do
    Begin
        Sy:=Sign(y[i]);
        Sz:=Sign(z[i]);
        If Sy=Sz then
            Inc(m);
        End;
Writeln('Кількість співпадаючих за знаком '+'
        'елементів m = ', m);

{ Припинення роботи програми }
WaitEscape;

End.

```

## ВАРІАНТИ ЗАВДАНЬ ДО ЛАБОРАТОРНОЇ РОБОТИ № 1

### "ОБРОБКА ОДНОВИМІРНИХ МАСИВІВ"

*Вказівка.* Порівняння дійсних чисел скрізь виконувати з урахуванням погрішності  $\varepsilon$  поза залежністю від того, чи є відповідна вказівка в тексті конкретного завдання.

*Примітка.* Для обчислення функцій, які не входять до складу бібліотеки Турбо Паскаля ( $\arcsin(x)$ ,  $\lg(x)$ ,  $sh(x)$  і ін.), можна використати наступні формули:

1)  $y = \arcsin(x)$

$$y = \begin{cases} \frac{\pi}{2}, & \text{якщо } ||x| - 1| < \varepsilon \\ \operatorname{arctg} \frac{x}{\sqrt{1-x^2}} & \text{в інших випадках} \end{cases}$$

Реалізація на Паскалі при  $\varepsilon = 10^{-10}$ :

```
if abs (abs (x) -1) <1E-10 then  
  y:=pi/2  
else  
  y:=arctan (x/sqrt (1-sqr (x) ) ) ;
```

2)  $y = \arccos(x)$

$$y = \begin{cases} \frac{\pi}{2}, & \text{якщо } |x| < \varepsilon \\ \operatorname{arctg} \frac{\sqrt{1-x^2}}{x}, & \text{якщо } x > 0 \\ \operatorname{arctg} \frac{\sqrt{1-x^2}}{x} + \pi, & \text{якщо } x < 0 \end{cases}$$

Реалізація на Паскалі:

```
if abs (x) <1E-10 then  
  y:=pi/2  
else  
  y:=arctan (sqrt (1-sqr (x) ) /x) +pi*byte (x<0) ;
```

Операнд `byte (x<0)` означає, що логічний вираз  $x < 0$  (його значенням може бути *true* або *false*, внутрішнє подання - однобайтне значення 1 або 0) розглядається як ціла змінна типу `byte` і може мати відповідно значення 1 або 0.

Примітка.  $\text{Arcsin}(x) = \frac{\pi}{2}$ , якщо модуль аргументу  $x$  перебуває в  $\varepsilon$ -околиці значення 1,

тобто  $1 - \varepsilon < |x| < 1 + \varepsilon$ .  $\text{Arccos}(x) = \frac{\pi}{2}$ , якщо модуль аргументу  $x$  перебуває в  $\varepsilon$ -околиці значення 0,

тобто  $-\varepsilon < x < +\varepsilon$ .

3)  $y = \text{arcctg}(x)$

$$y = \frac{\pi}{2} - \text{arctg}(x)$$

4)  $y = \text{lg}(x)$

$$y = \frac{\ln(x)}{\ln(10)}$$

5)  $y = \text{sh}(x)$  - гіперболічний синус:  $y = \frac{e^x - e^{-x}}{2}$ ;

6)  $y = \text{ch}(x)$  - гіперболічний косинус:  $y = \frac{e^x + e^{-x}}{2}$ ;

7) Гіперболічні тангенс, котангенс, секанс і косеканс ( $\text{th}(x)$ ,  $\text{cth}(x)$ ,  $\text{sch}(x)$ ,  $\text{csch}(x)$ ) визначаються аналогічно відповідним тригонометричним функціям ( $\text{th}(x) = \text{sh}(x)/\text{ch}(x)$  і т.д.).

8)  $y = \text{Arsh}(x)$  - ареасинус:  $y = \ln\left(x + \sqrt{x^2 + 1}\right)$ ;

1. Знайти середнє арифметичне значення  $S$  елементів масиву  $Z$  і замінити додатні елементи  $z_i$  абсолютним значенням різниці  $S - z_i$ . Визначити, як при цьому змінилися позиції максимального й мінімального по модулю елементів масиву  $Z$ .

$$z_i = \begin{cases} \cos^2 x_i + \text{tg} \frac{x_i}{2} & , \text{ якщо } \frac{\pi}{8} < |x_i| < \frac{3}{4}\pi \\ \sin x_i^2 - \sqrt{x_i^2 + 1} & , \text{ в інших випадках} \end{cases}$$

2. Кожну пару елементів масиву  $Z$  розташувати в порядку зростання, тобто  $z_1 \leq z_2$ ,  $z_3 \leq z_4$ ,  $z_5 \leq z_6$ , ... Якщо масив  $Z$  містить непарну кількість елементів  $n$ , то останній елемент  $z_n$  не розглядається. Визначити, наскільки змінилася позиція максимального по модулю елемента масиву  $Z$  після його перетворення.

$$y_i = \frac{\pi}{3} \sin \frac{x_i}{2}; \quad z_i = \begin{cases} 1 + e^{0,5x_i - x_i^2} & , \text{якщо } y_i \geq 0,1 \\ 1 + x_i + \sqrt[3]{x_i^2 - 1} & , \text{якщо } y_i < 0,1 \end{cases}$$

3. Обчислити відстань  $L$  між двома  $n$ -мірними точками, координати яких задані векторами  $Y$  і  $Z$ , а також відстані  $d_y$  і  $d_z$  від кожної із цих точок до початку координат.

$$y_i = 2 \sin^2 \frac{\pi}{3} x_i + 0,5 \sqrt{|x_i^3|}; \quad z_i = \begin{cases} 1 + e^{-y_i} & , \text{якщо } y_i > 3 \\ y_i + \sqrt{|y_i|} & , \text{якщо } -1 \leq y_i \leq 3 \\ \cos y_i + \frac{1}{y_i^3} & , \text{якщо } y_i < -1 \end{cases}$$

*Примітка.*

$$L = \sqrt{\sum_{i=1}^n (y_i - z_i)^2}; \quad d_y = \sqrt{\sum_{i=1}^n y_i^2}; \quad d_z = \sqrt{\sum_{i=1}^n z_i^2}$$

4. Визначити, скільки в масиві  $Z$  є додатних чисел, які оточені ліворуч і праворуч принаймні одним від'ємним числом. Надрукувати значення й порядковий номер максимального з таких чисел.

$$y_i = 2\pi \sin^2(\pi x_i) + 5 \cos\left(\frac{\pi}{2} x_i^2\right); \quad z_i = \begin{cases} 2,5 y_i^{-3} - 6,8 y_i^{-1} + 0,8 & , \text{якщо } |y_i| > 3 \\ 1,5 e^{|y_i - 2|} - \sqrt{|y_i + 5|} & , \text{якщо } |y_i| \leq 3 \end{cases}$$

*Примітка.* Для елемента  $z_1$  «сусідом» ліворуч є елемент  $z_n$ , а для  $z_n$  сусідом праворуч вважається елемент  $z_1$ .

5. Елементи  $(x_i, y_i, z_i)$  - це компоненти тривимірного вектора. Визначити, чи є серед отриманих у такий спосіб векторів колінеарні (паралельні) вектори, і якщо такі випадки будуть виявлені, надрукувати номери першої пари колінеарних векторів.

$$y_i = \begin{cases} 1 + e^{\sin x_i} & , \text{якщо } |x_i| \geq \pi \\ \frac{1}{2} x_i^3 + \sqrt[3]{x_i} & , \text{якщо } |x_i| < \pi \end{cases}; \quad z_i = \sqrt{2 + \cos^2\left(\frac{\pi}{2} + y_i\right)} + 2 \sin \frac{y_i}{2}$$

*Вказівка.* Два вектори  $(x_1, y_1, z_1)$  і  $(x_2, y_2, z_2)$  колінеарні, якщо вони мають однакові напрямні косинуси. Для першого із зазначених вище векторів маємо:

$$\cos\alpha = \frac{x_1}{\sqrt{x_1^2 + y_1^2 + z_1^2}}; \quad \cos\beta = \frac{y_1}{\sqrt{x_1^2 + y_1^2 + z_1^2}}; \quad \cos\gamma = \frac{z_1}{\sqrt{x_1^2 + y_1^2 + z_1^2}}$$

Перевірку коллінеарності виконувати з урахуванням погрішності  $\varepsilon$ .

6. Визначити абсолютну і відносну (у відсотках) кількість елементів масиву  $P$ , які збігаються за знаком з відповідними елементами масиву  $X$ . При цьому вказати окремо кількість співпадаючих додатних, від'ємних і нульових елементів. Порівняння проводити з урахуванням припустимої погрішності  $\varepsilon$ .

$$y_i = \begin{cases} 1 + e^{x_i} \sin x_i & , \text{ якщо } |x_i| < 3 \\ 0,2x_i - \cos^2 x_i & , \text{ якщо } |x_i| \geq 3 \end{cases}; \quad p_i = 15,6y_i^5 - 32y_i^4 - 17y_i^3 + 81y_i - 159$$

*Вказівка.* Змінні  $p_i$  обчислювати за схемою Горнера.

7. Обчислити скалярний добуток  $S$  векторів  $Y$  й  $Z$  і визначити (у відсотках зі знаком), наскільки воно зміниться, якщо

- елементи вектора  $Z$  переставити у зворотному порядку;
- елементи вектора  $Z$  згрупувати в порядку зменшення.

$$y_i = 2,5 \sin \pi x_i - \sqrt{|x_i|}; \quad z_i = \begin{cases} \ln(|x_i| + 1) + \sqrt[3]{x_i} & , \text{ якщо } |y_i| > 0,5 \\ x_i^2 + x_i + 1 & , \text{ якщо } |y_i| \leq 0,5 \end{cases}$$

*Примітка.*

$$S = \sum_{i=1}^n y_i z_i$$

8. За отриманим значенням масивів  $Y$  й  $Z$  сформувати масив  $P$ , використовуючи такі відношення:

$$p_1 = y_1 + z_1; \quad p_2 = y_1 - z_1$$

$$p_3 = y_2 + z_2; \quad p_4 = y_2 - z_2$$

.....

$$y_i = \sin^2 x_i - \cos^2 x_i^2; \quad z_i = \begin{cases} \ln\left(\cos^2 \frac{\pi}{4} x_i + 1,5\right) & , \text{ якщо } y_i > 0,25 + \sin x_i \\ e^{1+x_i} - \sqrt[3]{y_i} & , \text{ якщо } y_i \leq 0,25 + \sin x_i \end{cases}$$

Визначити, чи є в масиві  $P$  елементи, які відрізняються від нуля не більше ніж на задане мале значення  $\varepsilon$ . Якщо такі елементи є, то обміняти перші два з них місцями.

9. Обчислити середнє арифметичне  $S$  і дисперсію  $D$  елементів масиву  $Y$ . Визначити відносну кількість (у відсотках) елементів масиву  $Y$ , які відрізняються від значення  $S$  більше ніж на значення  $D$ .

$$y_i = \begin{cases} \frac{\sqrt{x_i} \sin \pi x_i}{\ln|x_i| + e^{-x_i}} & , \text{якщо } x_i > 1,5 \\ 2\sqrt[3]{x_i + \ln|x_i| + 1} & , \text{якщо } x_i \leq 1,5 \end{cases}$$

$$S = \frac{1}{n} \sum_{i=1}^n y_i; \quad D = \frac{1}{n-1} \sum_{i=1}^n (y_i - S)^2$$

10. Знайти в масиві  $Z$  максимальний елемент  $z_{\max}$  і замінити в цьому масиві кожен елемент, менший ніж  $0,5 z_{\max}$ , його доповненням до  $z_{\max}$ . Визначити абсолютну та відносну (у відсотках) кількість таких замінів.

$$y_i = 2\sqrt[3]{x_i} + 0,1x_i - \frac{x_i}{|x_i|+1}; \quad z_i = \begin{cases} y_i - \frac{1}{3} \frac{y_i^2}{y_i + 1} & , \text{якщо } y_i > 1 \\ 0,5 \cos \pi y_i & , \text{якщо } |y_i| \leq 1 \\ 2 \sin \left( \cos \left( \frac{\pi}{2} y_i \right) \right) & , \text{якщо } y_i < -1 \end{cases}$$

11. Кількість елементів масиву  $X$  задати парним. Уважаючи елементи  $y_i$  абсцисами,  $z_i$  - ординатами на площині  $YOZ$ , а кожну пару точок  $(y_1, z_1) - (y_2, z_2)$ ,  $(y_3, z_3) - (y_4, z_4)$ , ... - координатами початку й кінця відрізка, визначити, скільки відрізків повністю розташовані в першому квадранті. Визначити також, між якими точками першого й останнього відрізків відстань максимальне.

$$y_i = \begin{cases} \frac{\pi}{2} \cos \left( \frac{x_i}{2} + \pi \right) - 2 & , \text{якщо } x_i > 1 \\ 2 \sin x_i \cdot \cos \left( \frac{x_i}{2} - \frac{\pi}{4} \right) & , \text{якщо } x_i \leq 1 \end{cases};$$

$$z_i = 2y_i^2 - \sqrt[5]{y_i + 1}$$

*Вказівка.* Приналежність точки квадранту визначати з урахуванням заданої погрішності  $\varepsilon$ .

12. Визначити суми й кількості додатних і від'ємних елементів вектора  $P$ . Елементи, які по модулю менші заданого значення  $\varepsilon$  (наприклад, 0.001), не враховувати. Після цього обміняти місцями перший додатний і перший від'ємний елементи масиву  $P$  (якщо відповідні елементи є в даному масиві).

$$y_i = 3 \sin^3 \frac{\pi}{3} x_i + 0,5 \ln \left( |x_i| + \pi \right) \quad z_i = \begin{cases} 1 - e^{-y_i} & , \text{якщо } y_i > 2 \\ \cos y_i - y_i^2 & , \text{якщо } -1 \leq y_i \leq 2; \\ y_i + 2\sqrt{|y_i|} & , \text{якщо } y_i < -1 \end{cases}$$

$$p_i = z_i + 0,4 \frac{x_i}{2 + \cos x_i}$$

13. Уважаючи елементи масиву  $Z$  довжинами відрізків, визначити, чи можна з відрізків (1, 2, 3), (2, 3, 4), (3, 4, 5), ... побудувати трикутники, і підрахувати кількість отриманих при цьому трикутників. Надрукувати номери відрізків, які утворюють трикутник найбільшого периметра.

*Вказівка.* Використати наступну властивість: напівпериметр трикутника більше довжини кожної з його сторін.

$$y_i = \begin{cases} 5e^{-x_i} + \frac{100 x_i}{1 + x_i} & , \text{якщо } x_i > 1 \\ 2 \cos^2 \left( \left| x_i \right| + \pi \right) & , \text{якщо } |x_i| \leq 1; \\ 2 \sin^2 \left( \cos \frac{\pi}{2} x_i \right) & , \text{якщо } x_i < -1 \end{cases}$$

$$z_i = \sqrt[5]{y_i + 1}$$

Можливість побудови трикутника визначати з урахуванням погрішності  $\varepsilon$ .

14. У масиві  $Y$  знайти значення й положення найбільшого від'ємного й найменшого додатного числа, після чого обміняти їх місцями.

$$y_i = \begin{cases} \frac{\pi}{3} \cos^2 \left( x_i - \frac{\pi}{2} \right) - 2 \sin x_i & , \text{якщо } 2 < x_i \leq 10 \\ 2e^{-x_i} - \ln \left( |x_i| + 1 \right) & , \text{якщо } x_i > 10 \\ x_i^3 + x_i^2 + x_i + 1 & , \text{в інших випадках} \end{cases}$$

*Вказівка.* Значення  $y_i$  в третій галузі обчислювати за схемою Горнера. Елементи, по



модулі менші ніж  $\varepsilon$ , не враховувати.

15. Елементом масиву  $Y$ , які відрізняються від середнього арифметичного значення  $S$  цього масиву не більше ніж на  $\varepsilon$ , привласнити нульове значення. Визначити, на скільки відсотків змінилося при цьому значення  $S$ .

$$y_i = \begin{cases} \frac{\sqrt{|x_i|} \left( 2 + \sin \left( x_i + \frac{\pi}{2} \right) \right)}{\ln|x_i| + e^{-|x_i|}} & , \text{якщо } |x_i| > 5 \\ 2x_i - \sqrt[3]{x_i} & , \text{якщо } |x_i| \leq 5 \end{cases}$$

16. Визначити порядкові номери й значення перших позитивних  $P_z$  й останнього негативного  $N_z$  елементів масиву  $Z$ , після чого обчислити середнє арифметичне значення елементів, які позиційно розташовані між  $P_z$  й  $N_z$ .

$$y_i = \sin^2 x_i + sh \frac{x_i}{2}; \quad z_i = \begin{cases} \ln \left( \cos^2 \frac{\pi}{4} x_i + 1 \right) + 1 & , \text{якщо } |y_i| > x_i^2 \\ 1 + x_i - x_i^2 + x_i^3 - x_i^4 & , \text{якщо } |y_i| \leq x_i^2 \end{cases}$$

*Вказівка.* Обчислення елементів  $z_i$  при  $|y_i| \leq x_i^2$  виконувати за схемою Горнера. При переборі позитивних і негативних елементів масиву  $Z$  ті з них, які по модулі менше ніж  $\varepsilon$ , не враховувати, де  $\varepsilon$  - мала величина (наприклад, 0.001).

17. Елементи  $(x_i, y_i)$  - це координати точок на площині. Розглядаючи пари точок  $(1, 2)$ ,  $(2, 3)$ ,  $(3, 4)$ , ... як початок і кінець відрізка, визначити, скільки таких відрізків однократно перетинає коло радіусом  $R$  із центром у точці  $(x_c, y_c)$ . Значення  $R$ ,  $x_c$  і  $y_c$  ввести із клавіатури.

$$y_i = \begin{cases} 2 \left( \cos^2 x_i - 1 \right) + \ln^2 \left( x_i^2 + 1 \right) & , \text{якщо } |x_i| > 2 \\ 2 \left( sh x_i^2 + 1 \right) + 3\sqrt[3]{x_i} & , \text{якщо } |x_i| \leq 2 \end{cases}$$

18. Визначити суму й добуток тих елементів масиву  $P$ , для яких виконується відношення  $p_i > 2x_i$ , а також середнє арифметичне значення таких елементів.

$$P_i = \begin{cases} 2 \sin \pi x_i^2 + \sqrt[3]{x_i} \cdot \cos\left(x_i - \frac{\pi}{3}\right) & , \text{ якщо } |x_i| > \frac{\pi}{3} \\ 2 \operatorname{sh}\left(x_i + \frac{\pi}{6}\right) - e^{x_i} \sqrt{|x_i| + 1} & , \text{ якщо } |x_i| \leq \frac{\pi}{3} \end{cases}$$

19. Визначити значення й порядкові номери елементів масиву  $Y$ , які в найбільшому й найменшому ступені відрізняються від середнього арифметичного значення  $S$  цього масиву, після чого обміняти їх місцями.

$$y_i = \begin{cases} 2,8 \sin \sqrt{\ln^2 x_i + 1} + 1 & , \text{ якщо } x_i > 1 \\ \sqrt[3]{e^{x_i} - 2} + x_i \operatorname{sh} x_i & , \text{ якщо } x_i \leq 1 \end{cases}$$

*Примітка.* У завданні мова йде про знаходження таких елементів  $y_i$ , для яких вираз  $|y_i - S|$  має відповідно максимальне або мінімальне значення.

20. Уважаючи елементи  $(x_i, y_i)$  абсцисами й ординатами точок на площині, виконати перетворення координат шляхом повороту осей таким чином, щоб вісь абсцис пройшла через точку  $(x_2, y_2)$ . Визначити, чи змінився номер точки, найближчої до початку координат, після такого перетворення.

$$x_i = \pi \sin^2\left(t_i - \frac{\pi}{2}\right) + 1; \quad y_i = \begin{cases} 0,8\sqrt{|t_i| + 1} - t_i & , \text{ якщо } |t_i| < 2 \\ 0,3\sqrt[3]{|t_i| + 1} - 0,1e^{0,1t_i} & , \text{ якщо } |t_i| \geq 2 \end{cases}$$

*Вказівка.*

Відстань від точки  $(x_i, y_i)$  до початку координат визначається по формулі

$$d = \sqrt{x_i^2 + y_i^2}$$

Координати точок у новій системі

$$\begin{aligned} x' &= x \cos \alpha + y \sin \alpha \\ y' &= -x \sin \alpha + y \cos \alpha \end{aligned}$$

де

$$\sin \alpha = \frac{y_2}{\sqrt{x_2^2 + y_2^2}}; \quad \cos \alpha = \frac{x_2}{\sqrt{x_2^2 + y_2^2}}$$

Оскільки змінні  $x$  і  $y$  входять у формули для  $x'$  і  $y'$ , те обчислення нових координат робити в такий спосіб:

$$R = \sqrt{x_2^2 + y_2^2}; \quad \sin \alpha = \frac{y_2}{R}; \quad \cos \alpha = \frac{x_2}{R}$$

$$R = x_i \cos \alpha + y_i \sin \alpha; \quad y_i = -x_i \sin \alpha + y_i \cos \alpha$$

$$x_i = R$$

21. Обміняти місцями максимальний  $y_{\max}$  і мінімальний  $y_{\min}$  елементи масиву  $Y$ , після чого визначити середнє арифметичне значення  $S$  всіх елементів масиву  $Y$ , за винятком  $y_{\max}$  і  $y_{\min}$ . Визначити кількість елементів масиву  $Y$ , які відрізняються від значення  $S$  не більше ніж на 10 %.

$$y_i = \begin{cases} \frac{1,5e^{-x_i}}{|a|+1} + b \left| \sin \left( x_i - \frac{\pi}{3} \right) \right| & , \text{ якщо } x_i \geq 1 \\ 6,5 x_i + 0,5 \sqrt[3]{x_i + 1} - \operatorname{sh} x_i & , \text{ якщо } x_i < 1 \end{cases}$$

*Вказівка.* Значення коефіцієнтів  $a$  й  $b$  увести із клавіатури.

22. Визначити степінне середнє  $S$  модулів елементів масиву  $P$ , а також значення і порядкові номери елементів  $p_i$ , які в найбільшій і найменшій мірі відрізняються від значення  $S$ , після чого обміняти місцями ці елементи.

$$y_i = \begin{cases} 0,5 \left( \ln^2 x_i + 1 \right) + \sqrt{|x_i|} \sin x_i & , \text{ якщо } |x_i| < 5 \\ 2 \ln |x_i| + 1 + x_i^2 + 1 & , \text{ якщо } |x_i| \geq 5 \end{cases}$$

$$p_i = 3,4 y_i^7 - 2,9 y_i^5 - 3,8 y_i^4 + 6,7 y_i^2 - 2,4 y_i + 190$$

*Вказівка.* Обчислення елементів  $p_i$  робити за схемою Горнера.

*Примітка.* Степінним середнім додатних чисел  $p_1, p_2, \dots, p_n$  називається величина

$$S = \left( \frac{1}{n} \sum_{i=1}^n p_i^m \right)^{\frac{1}{m}}$$

де  $m$  - будь-яке ціле додатне число (наприклад,  $m = 3$ ).

Значення  $m$  повинне бути уведене із клавіатури.

23. Знайти довжину кривої  $y(x)$  як суму довжин відрізків, координати кінцевих точок яких визначаються елементами  $(x_i, y_i)$ . Надрукувати порядкові номери самого довгого й самого короткого відрізка.

$$y_i = \begin{cases} \frac{10\sqrt[3]{x_i+1} + 2 \sin x_i}{\ln(|x_i|+1)+1} & , \text{ якщо } |x_i| > 1 \\ 10\sqrt{|x_i+1|} + 0,5 \operatorname{sh} x_i & , \text{ якщо } |x_i| \leq 1 \end{cases}$$

24. Обчислити середнє арифметичне  $S$  масиву  $Y$  і визначити кількість елементів  $y_i$ , які відрізняються від  $S$  не більше ніж на 30%. Знайти серед цих елементів мінімальний і максимальний по модулю, після чого обміняти їх місцями.

$$y_i = \begin{cases} 0,5 \left( 2 + \sin^2 \left( x_i - \frac{\pi}{4} \right) \right) + e^{x_i} & , \text{ якщо } x_i < 1 \\ 4x_i^{0,62} - 2\sqrt{2x_i+1} + 1 & , \text{ якщо } 1 \leq x_i \leq 10 \\ 5e^{-x_i} + 2 \ln^2 x_i & , \text{ якщо } x_i > 10 \end{cases}$$

25. Обчислити середнє арифметичне значення  $S$  і середнє квадратичне відхилення  $\sigma$

$$y_i = \begin{cases} 2 + \pi x_i \sin \left( x_i - \frac{\pi}{4} \right) & , \text{ якщо } x_i \leq 0 \\ 1 + \cos^2 \left( \frac{\pi}{2} x_i \right) + x_i & , \text{ якщо } 0 < x_i \leq 10 \\ x_i - \sqrt{\ln x_i} + e^{-x_i} & , \text{ якщо } x_i > 10 \end{cases}$$

$$S = \frac{1}{n} \sum_{i=1}^n y_i; \quad \sigma = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (y_i - S)^2}$$

Визначити абсолютну і відносну кількість елементів  $y_i$ , які лежать в інтервалі  $S \pm \sigma$  і поза інтервалом  $S \pm 2\sigma$  (окремо по кожному інтервалу).

26. Уважаючи  $(x_i, y_i)$  координатами точок на площині, знайти відстань між двома точками, одна із яких найбільш віддалена від осі абсцис, а друга - від осі ординат. Якщо ці точки не збігаються, то обміняти їх місцями.

$$x_i = 2\pi \sin \left( t_i + \frac{\pi}{3} \right) - \cos \left( 2t_i - \frac{\pi}{3} \right); \quad y_i = \begin{cases} 0,5 e^{0,5t_i} - x_i^2 - 1 & , \text{ якщо } |x_i| \geq \pi \\ 2\sqrt[3]{x_i} + x_i + 1 & , \text{ якщо } |x_i| < \pi \end{cases}$$

27. Сформувати вектор  $C$  по формулі  $c_i = \max(a_i, b_i) - \min(a_i, b_i)$  і знайти в ньому максимальне значення серед елементів з парними номерами й мінімальне – серед елементів з непарними номерами, після чого обміняти місцями ці елементи.

$$a_i = \sqrt{\left| \sin \left( x_i + \frac{\pi^2}{10} \right) + \cos \frac{\pi^2}{10} x_i \right|}; \quad b_i = \frac{\sqrt{e^{\lg(|x_i|+1)} + \sin^2 x_i}}{1 + \lg(|x_i|+10)}$$

28. Задавши кількість елементів масиву  $X$  парним і вважаючи елементи  $x_i$  абсцисами, а  $y_i$  - ординатами точок на площині, визначити, скільки відрізків  $(x_1, y_1) - (x_2, y_2)$ ,  $(x_3, y_3) - (x_4, y_4)$ , ... перетинають вісь  $X$  і скільки з них перетинають вісь  $Y$ . Відрізки, у яких одна із точок лежить на осі  $X$  або на осі  $Y$ , не враховувати.

$$u_i = 0,5 \sin(0,5x_i) + \frac{\pi}{2} \cos \left( x_i + \frac{\pi}{2} \right); \quad y_i = \begin{cases} \sin(\sin u_i) & , \text{якщо } u_i > 0 \\ \cos \left( 1 + e^{\sqrt{1-u_i}} \right) & , \text{якщо } u_i \leq 0 \end{cases}$$

29. Стосовно масиву  $Z$  виконати наступні дії:

- якщо в масиві  $Z$  є нульові елементи, то замінити їх напівсумою суміжних елементів; для першого елемента суміжним ліворуч вважати останній елемент, для останнього елемента суміжним праворуч є перший елемент;

- знайти максимальний  $z_{\max}$  і мінімальний  $z_{\min}$  по модулю елементи;

- обчислити середнє геометричне модулів всіх елементів масиву  $Z$ , за винятком  $z_{\max}$  і  $z_{\min}$ .

$$y_i = 15,6x_i^5 - 32x_i^4 - 17x_i^3 + 49x_i^2 + 81x_i - 159$$

$$z_i = \begin{cases} 1 + \sqrt{1 + 0,5 \cos^2 y_i} & , \text{якщо } y_i < 0 \\ 2 + \sqrt{1 + e^{-0,5y_i}} & , \text{якщо } y_i > 10 \\ 0 & , \text{в інших випадках} \end{cases}$$

*Вказівка.* Обчислення елементів  $y_i$  робити за схемою Горнера.

*Примітка.* Середнє геометричне обчислюється по формулі:

$$P = \sqrt[n]{\prod_{i=1}^n |z_i|}$$

30. По масиві  $Y$  сформувати масив  $Z$ , у якому спочатку розташувати всі від'ємні елементи  $y_i$ , а після цього - всі невід'ємні елементи цього масиву, зберігши їх початковий відносний порядок. Визначити, як при цьому змінилося положення мінімального елемента масиву.

$$x_i = \begin{cases} e^{\sin t_i} - 1 & , \text{ якщо } t_i \leq 2 \\ \sqrt{t_i + 1} - 2 & , \text{ якщо } t_i > 2 \end{cases}; \quad y_i = 0,9x_i^8 - 12x_i^7 + 8x_i^5 - 3x_i^4 + 4x_i^2 - 19,3$$

*Вказівка.* Обчислення змінної  $y_i$  виконувати за схемою Горнера.

*Примітка.* При формуванні масиву  $Z$  дозволяється використати буферний масив.

31. Вважаючи елементи  $(y_i, z_i)$  координатами точок на площині  $YOZ$ , визначити кількість і відсоток точок, для яких кожне з відстаней до двох суміжних точок не перевищує значення параметра  $d$ , що вводиться із клавіатури. Для першої точки суміжною ліворуч вважати останню точку, для останньої точки суміжною праворуч є перша точка.

$$z_i = \begin{cases} 2 \sin^2 \left( y_i - \frac{\pi}{2} \right) + 0,5 \operatorname{sh}(y_i) & , \text{ якщо } |y_i| < \frac{\pi}{2} \\ 3y_i^3 + 2y_i^2 - 5y_i - 12 & , \text{ якщо } |y_i| \geq \frac{\pi}{2} \end{cases}$$

32. Вважаючи елементи  $(x_i, y_i)$  координатами точок на площині, визначити кількість і відсоток п'ячок, розташованих усередині кільця з радіусами  $R$  і  $r$  та центром у точці  $(x_c, y_c)$ . Значення  $R, r, x_c, y_c$  увести із клавіатури. Якщо при уведенні має місце  $R < r$ , то обміняти значення цих змінних.

$$y_i = \begin{cases} 2 \ln \left( x_i^2 + 1 \right) + \sin^2 x_i^2 + \sqrt[3]{x_i} & , \text{ якщо } x_i < 1 \\ \frac{x_i \sqrt{\sin^2 x_i + \cos^2 x_i^2 + 1}}{\sqrt{2x_i + 1} \ln \left( x_i + 1 \right) + 1} & , \text{ якщо } x_i \geq 1 \end{cases}$$

33. Визначити кількість елементів масиву  $Y$ , що задовольняють умові  $y_i \in [a, b]$ , де значення параметрів  $a$  і  $b$  вводяться із клавіатури. Якщо при уведенні має місце  $a > b$ , то обміняти значення цих параметрів. Визначити максимальне й мінімальне значення серед елементів, які задовольняють наведеній умові, після чого обміняти місцями ці елементи.

$$y_i = \begin{cases} x_i^4 + \frac{2x_i^2 + 3}{\pi + x_i} \sin x_i + 0,47 & , \text{ якщо } x_i > 1 \\ x_i \sin \sqrt[3]{x_i} + \frac{2 \ln(|x_i| + 1) + 1}{1 + 2 \cos^2 \left( x_i - \frac{\pi}{3} \right)} & , \text{ якщо } x_i \leq 1 \end{cases}$$

34. У масиві  $Y$  обміняти місцями елементи кожної пари чисел  $(y_i, y_{i+1})$ ,  $i = 1, 3, 5, \dots$ . Якщо кількість елементів  $n$  у масиві  $Y$  непарна, то останній елемент в обміні не бере участь. Визначити, наскільки при цьому змінилася кількість суміжних пар  $(y_i, y_{i+1})$ ,  $i = 1, 2, \dots, n-1$ , для яких виконується відношення  $y_i > y_{i+1}$ .

$$y_i = \begin{cases} 2,8 \sin \frac{x_i}{2} + \operatorname{tg} x_i + \sqrt{|x_i|} & , \text{ якщо } |x_i| < 0,45\pi \\ 2,8 \operatorname{sh} \frac{x_i}{2} + \operatorname{th} x_i + \sqrt[3]{x_i} & , \text{ якщо } |x_i| \geq 0,45\pi \end{cases}$$

35. Вважаючи елементи  $(x_i, y_i)$  координатами кінцевих точок відрізків ламаної лінії, визначити номер відрізка, що перетинає вісь абсцис, а також координати точки перетинання. Якщо таких відрізків декілька, то вказати номер найкоротшого з них.

$$y_i = \begin{cases} 2 \sin^3 2x_i - \cos \left( x_i - \frac{\pi}{6} \right) & , \text{ якщо } |x_i| < 1 \\ \cos \sqrt[3]{x_i} - 2 \sin \frac{x_i}{2} & , \text{ якщо } |x_i| \geq 1 \end{cases}$$

36. Обчислити середнє арифметичне  $S$  і середнє геометричне  $P$  елементів масиву  $M$ , після чого визначити кількість елементів  $m_i$ , розташованих у діапазоні від  $\min(S, P)$  до  $\max(S, P)$ .

$$m_i = \begin{cases} \left| 1 + \operatorname{arctg} \frac{x_i}{1 + \sqrt{x_i}} - x_i \right| & , \text{ якщо } x_i > 0,15 \\ \left| \sin x_i \cdot e^{x_i} - x_i \right| + 1 & , \text{ якщо } x_i \leq 0,15 \end{cases}$$

*Примітка.* Середнє геометричне обчислюється по формулі:

$$P = n \sqrt[n]{\prod_{i=1}^n |m_i|}$$

37. Вважаючи елементи  $(x_i, y_i)$  координатами точок на площині, знайти найкоротшу відстань між двома суміжними точками. При цьому першу й останню точки також вважати суміжними. Надрукувати порядкові номери цих точок і значення відстані між ними.

$$y_i = \begin{cases} 0,5e^{0,5x_i} + 0,5sh(0,5x_i) & , \text{ якщо } |x_i| < 3 \\ 2 \sin \sqrt[3]{x_i + 2} + \ln(|x_i| + 1) & , \text{ якщо } |x_i| \geq 3 \end{cases}$$

38. Вважаючи елементи  $(x_i, y_i)$  координатами точок на площині, визначити, чи є в масивах  $X, Y$  три суміжні точки  $(1, 2, 3), (2, 3, 4), (3, 4, 5), \dots$ , які лежать на одній прямій. Підрахувати кількість таких випадків і надрукувати номери першої тріади цих точок.

$$y_i = \begin{cases} 0,5x_i^3 - x_i^2 + x_i - 1 & , \text{ якщо } |x_i| \leq 1 \text{ или } 3 \leq |x_i| \leq 4 \\ \sin^2\left(x_i - \frac{\pi}{4}\right) & , \text{ якщо } 1 < x_i < 3 \\ \cos^3 x_i^2 & , \text{ якщо } -3 \leq x_i \leq -1 \\ 2 + x_i & , \text{ якщо } 4 < x_i \leq 5 \\ -2 + 5x_i & , \text{ якщо } x_i > 5 \\ 0 & , \text{ якщо } x_i < -3 \end{cases}$$

*Вказівка.* Три точки  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$  лежать на одній прямій, якщо

$$d = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} = 0 \Rightarrow x_1(y_2 - y_3) - x_2(y_3 - y_1) + x_3(y_1 - y_2) = 0$$

Рівність визначника  $d$  нулю перевіряти з точністю  $\varepsilon$ .

39. Елементи  $(x_i, y_i, z_i)$  є компонентами тривимірного вектора. Визначити, чи є серед отриманих у такий спосіб векторів два суміжних взаємно перпендикулярних вектори, і якщо такі випадки будуть виявлені, надрукувати кількість пар і номери першої пари перпендикулярних векторів.

$$y_i = 2 \sin^2\left(x_i - \frac{\pi}{6}\right) + \cos\left(\frac{x_i}{2} + \frac{\pi}{4}\right); \quad z_i = \begin{cases} e^{x_i} + 2\sqrt[3]{x_i} & , \text{ якщо } |y_i| < 0,5 \\ 3\sqrt{|x_i + y_i|} - sh y_i & , \text{ якщо } y_i > 1 \\ 0 & , \text{ в інших випадках} \end{cases}$$

*Вказівка.* Два вектори  $(x_1, y_1, z_1)$  і  $(x_2, y_2, z_2)$  перпендикулярні, якщо

$$d = x_1x_2 + y_1y_2 + z_1z_2 = 0 .$$



Перевірку перпендикулярності виконувати з урахуванням погрішності  $\varepsilon$ .

40. Визначити відносну кількість (у відсотках) елементів масиву  $Z$ , розташованих у діапазоні  $z_{\min} + (z_{\max} - z_{\min})/3 \dots z_{\min} + 2 \cdot (z_{\max} - z_{\min})/3$ , де  $z_{\min}$ ,  $z_{\max}$  - відповідно мінімальний і максимальний елементи масиву  $Z$ . Визначити також середнє арифметичне значення елементів, розташованих у зазначеному діапазоні.

$$z_i = \begin{cases} 1,3 \cos \sqrt{\text{abs}(x_i)} + e^{-0,1 x_i^2} - \pi x_i & , \text{ якщо } |x_i| \geq \frac{\pi}{2} \\ \sqrt{\text{arctg}|x_i| + 1} + \text{sh} \frac{x_i}{3} & , \text{ якщо } |x_i| < \frac{\pi}{2} \end{cases}$$

41. Елементи  $(x_i, y_i)$  - це координати центра, елемент  $R_i$  - радіус  $i$ -ого кола. Визначити, скільки суміжних кіл перетинаються між собою, стикаються або розміщені одне усередині іншого. Перше і останнє коло також вважати суміжними. Перевірку взаємного розташування кіл виконувати з урахуванням припустимої погрішності  $\varepsilon$ .

$$y_i = \begin{cases} 12,5 \sin \frac{\pi}{2} x_i + \text{sh} \frac{x_i}{2} & , \text{ якщо } |x_i| < 2\pi \\ 2,5 \cos \sqrt[3]{x_i} + \cos x_i & , \text{ якщо } |x_i| \geq 2\pi \end{cases}; \quad R_i = x_i^2 + |x_i| + 1$$

42. Елементи  $(x_i, y_i)$  - це координати точок на площині. Розглядаючи кожні три суміжні точки (1, 2, 3), (2, 3, 4), (3, 4, 5), ... як вершини трикутника, визначити номери трикутників, для яких радіус  $r$  вписаного в них кола має відповідно максимальне й мінімальне значення.

$$y_i = \begin{cases} 0,5 \lg (x_i^2 + 1) e^{-|x_i|} & , \text{ якщо } |x_i| > 2 \\ \pi \sin^2 \pi x_i - \text{sh} \left( \frac{x_i}{2} \right) & , \text{ якщо } |x_i| \leq 2 \end{cases}$$

Для визначення значення  $r$  використати формулу

$$r = \sqrt{\frac{(p-a)(p-b)(p-c)}{p}}$$

де  $a, b, c$  - довжини сторін трикутника,  $p$  - його напівпериметр.

43. Обміняти місцями перший від'ємний і останній додатний елементи масиву  $Z$ . Визначити також середні арифметичні значення окремо для додатних і від'ємних елементів масиву  $Z$ . Числа, які не перевищують по модулю значення  $\varepsilon$ , не враховувати.

$$z_i = \begin{cases} \frac{x_i \sin\left(x_i + \frac{\pi}{2}\right)}{1 + 0,5x_i^2} & , \text{ якщо } |x_i| < \frac{\pi}{4} \\ \sqrt[3]{\frac{\cos x_i + 3x_i}{|x_i| + 1}} & , \text{ якщо } |x_i| \geq \frac{\pi}{4} \end{cases}$$

44. Сформувати масив  $P$  по формулам

$$\begin{aligned} p_1 &= y_1 - 2x_1; & p_2 &= y_1 + x_1/2; \\ p_3 &= y_2 - 2x_2; & p_4 &= y_2 + x_2/2; \\ & \dots \dots \dots \end{aligned}$$

Визначити, на скільки відсотків відрізняються значення  $p_{\max}$  від суми  $x_{\max} + y_{\max}$ , а значення  $p_{\min}$  – від суми  $x_{\min} + y_{\min}$ , де  $x_{\max}$ ,  $y_{\max}$ ,  $p_{\max}$ ,  $x_{\min}$ ,  $y_{\min}$ ,  $p_{\min}$  – відповідно максимальні й мінімальні значення масивів  $X$ ,  $Y$  і  $P$ .

$$y_i = \begin{cases} 10x_i^{-3} + 2x_i^2 + \cos^3|x_i| & , \text{ якщо } |x_i| > 1 \\ \sqrt[5]{x_i^3 - 1} + e^{\sin x_i} & , \text{ якщо } |x_i| \leq 1 \end{cases}$$

45. Визначити, скільки разів у масиві  $P$  порушується впорядкованість його елементів по зростанню, тобто не виконується відношення  $p_i \leq p_{i+1}$ .

Обчислити середнє арифметичне значення елементів, для яких виконується відношення  $p_i \leq p_{i+1}$ .

$$p_i = \begin{cases} 0,5sh \frac{x_i}{2} - ch \frac{x_i^2}{2} + 1 & , \text{ якщо } |x_i| < 2 \\ \ln \frac{\sqrt[3]{x_i^2 + 1}}{|x_i| + 1} - \frac{\pi}{2} \sin \frac{x_i}{2} & , \text{ якщо } |x_i| \geq 2 \end{cases}$$

46. Елементи  $(x_i, y_i)$  – це координати точок ламаної лінії. Визначити, скільки відрізків ламаної, починаючи із другого, паралельні її першому відрізку, а також номер найбільш довгого з таких відрізків.

$$y_i = \begin{cases} \frac{x_i^2 - \sin^2 x_i}{\sqrt{|x_i|+1}} + e^{-x_i^2} & , \text{ якщо } |x_i| \leq 1 \\ e^{-|x_i|} \sin x_i + \ln(x_i^2 + 1) & , \text{ якщо } |x_i| > 1 \end{cases}$$

Вказівка. Дві прямі

$$a_1x + b_1y + c_1 = 0$$

$$a_2x + b_2y + c_2 = 0$$

паралельні, якщо

$$\Delta = a_1b_2 - a_2b_1 = 0 .$$

Для прямої, що проходить через точки  $(x_1, y_1)$  й  $(x_2, y_2)$ , коефіцієнти її загального рівняння визначаються по формулам:  $a = y_2 - y_1$ ;  $b = x_1 - x_2$ . Перевірку паралельності робити з точністю  $\varepsilon$ .

47. У масиві  $Y$  знайти останній від'ємний елемент, ціла частина якого більше одиниці і має непарне значення, і обміняти його місцями з першим додатним елементом, ціла частина якого більше одиниці і має парне значення.

$$y_i = \begin{cases} 5 \sin\left(x_i + \frac{\pi}{3}\right) \operatorname{sh}\left(x_i + \frac{\pi}{3}\right) + 1 & , \text{ якщо } x_i < \frac{\pi}{3} \\ 8e^{-2x_i} \cos(x_i - 1) \sqrt[3]{x_i^2} & , \text{ якщо } x_i \geq \frac{\pi}{3} \end{cases}$$

48. Елементи  $(x_i, y_i)$  - це абсциси й ординати точок на площині. Вважаючи кожні три суміжні точки  $(1, 2, 3)$ ,  $(2, 3, 4)$ ,  $(3, 4, 5)$ , ... вершинами трикутника, визначити порядкові номери трикутників, для яких радіус  $R$  описаного кола має відповідно мінімальне й максимальне значення.

$$y_i = \begin{cases} 3,5 \cos \frac{\pi x_i}{2} + \ln(x_i^2 + 1) & , \text{ якщо } |x_i| > \frac{\pi}{6} \\ 0,5 \operatorname{ch} \frac{\pi x_i}{2} + \sqrt[3]{x_i + 1} & , \text{ якщо } |x_i| \leq \frac{\pi}{6} \end{cases}$$

Вказівка. Для обчислення радіуса  $R$  використати формулу

$$R = \frac{abc}{4\sqrt{p(p-a)(p-b)(p-c)}}$$

де  $a, b, c$  – довжини сторін трикутника;  $p$  - його напівпериметр.

49. Вважаючи елементи  $(x_i, y_i)$  координатами кінцевих точок відрізків ламаної лінії, визначити номер відрізка, що перетинає вісь ординат, а також координати точки перетинання. Якщо таких відрізків кілька, то вказати номер найбільш довгого з них.

$$y_i = 10 \sin^2 \sqrt{|t_i|+1} - sh \frac{t_i}{3}; \quad x_i = \begin{cases} 2 \cos \left( t_i - \frac{\pi}{3} \right) + \sqrt[3]{t_i - 1} & , \text{ якщо } |t_i| > 2 \\ 3t_i^4 + 5t_i^3 + 8t_i^2 + t_i + 1 & , \text{ якщо } |t_i| \leq 2 \end{cases}$$

50. Визначити середнє арифметичне  $S$  елементів масиву  $Y$ , а також номери елементів, які в максимальному й мінімальному ступені відрізняються від значення  $S$ , після чого обміняти місцями ці елементи.

$$y_i = \begin{cases} \sqrt[3]{x_i + 1} + \ln(x_i^2 + 1) e^{-x_i^2} & , \text{ якщо } |x_i| > \pi \\ \frac{sh(x_i^2) + \sin(x_i^2) + 1}{x_i^2 + 1} & , \text{ якщо } |x_i| \leq \pi \end{cases}$$

## **Контрольна робота № 1**

### **ОБРОБКА МАТРИЦЬ**

#### Методичні вказівки

Мета роботи – практично освоїти методи розробки алгоритмів і методи програмної реалізації завдань обробки багатовимірних масивів.

У кожному з варіантів завдань до контрольної роботи № 1 необхідно скласти програму, що передбачає:

- уведення з текстового файлу елементів матриці;
- вивід уведених даних на принтер і (або) екран;
- обробку матриці відповідно до умови завдання;
- вивід результатів.

Вивід результатів потрібно організувати як на екран дисплея, так і на принтер. Для організації виводу рекомендується використати процедури `ScreenMatrix` і `PrinterMatrix`, наведені в прикладі виконання контрольної роботи.

Уведення матриці з текстового файлу, на відміну від одновимірного масиву, не можна організувати по ознаці кінця файлу (за значенням функції `SeekEof`). При уведенні з

використанням зазначеної ознаки матриці  $m \times n$  елементів програма може визначити лише загальну кількість елементів матриці, але вона не в змозі окремо вказати значення параметрів  $m$  і  $n$ . Тому рекомендується при формуванні текстового файлу в першому його рядку вказати значення  $m$  і  $n$  (для квадратної матриці – тільки значення  $n$ ), а в інших рядках – значення елементів матриці. Після уведення з файлу змінних  $m$  і  $n$  уведення елементів матриці виконується в циклі **For**, як це зроблено в прикладі.

Оброблювана в завданні матриця повинна мати від 5 до 10 рядків і стовпців.

Загальне зауваження. При рішенні ряду завдань варто звертати увагу на можливу невідповідність типів елементів масиву й вихідних результатів. Наприклад, середнє арифметичне значення, середнє квадратичне (див.п.1), дисперсія (див.п.27), інші види середніх значень, як правило, завжди дійсні поза залежністю від того, чи є цілочисельними або дійсними елементи оброблюваного масиву.

### **Пояснювальна записка по контрольній роботі №1**

У пояснювальній записці по контрольній роботі необхідно привести:

- титульний аркуш;
- номер варіанта і умову завдання;
- короткий опис програми;
- контрольний прорахунок (якщо результати роботи програми не є очевидними);
- текст програми;
- результати роботи програми.

Формат титульного аркуша - див. методичні вказівки до лабораторної роботи №1.

Друк тексту програми рекомендується виконувати за допомогою редактора мови Турбо Паскаль. Якщо ж для цього використовується система Word, то друк повинен виконуватися шрифтом Courier New. У цьому випадку всі виведені символи мають однакову ширину, чим забезпечується структурність друку програми. Для інших розділів звіту доцільно використовувати шрифт Times New Roman.

### **Приклад виконання завдання**

*Умова завдання.* У заданій квадратній матриці з дійсними елементами зрушити кожен її рядок циклічно вліво або вправо таким чином, щоб максимальний елемент рядка був розташований на головній діагоналі. Обчислити слід матриці до і після її перетворення.

Циклічне зрушення рядка припускає таку перестановку її елементів, при якій не порушується їх початкове відносне розташування.

Нехай, наприклад, третій рядок матриці має вигляд

5.8 -4.4 10.0 8.7 14.6 11.9

Тут максимальний елемент  $a_{3,5}=14.6$  розташований праворуч на дві позиції від елемента головної діагоналі  $a_{3,3}=10.0$ . Отже, необхідно двічі виконати циклічне зрушення рядка вліво, після чого він прийме вигляд

10.0 8.7 14.6 11.9 5.8 -4.4

Слід матриці - це сума елементів її головної діагоналі.

*Примітка.* Головну, як і побічну діагональ може мати тільки квадратна матриця.

Завдання, що сформульоване як приклад його виконання, реалізовано в наведеній нижче програмі Cont2.

При обробці матриці в кожному її рядку визначається максимальний елемент  $A_{max}$  і його порядковий номер (індекс)  $j_{max}$ , після чого обчислюється параметр  $k=i-j_{max}$ , де  $i$  - номер рядка матриці. Цей параметр вказує на положення елемента  $A_{max}$  щодо головної діагоналі матриці. Якщо  $k > 0$  (максимальний елемент перебуває ліворуч від елемента головної діагоналі), то виконується циклічне зрушення рядка вправо доти, доки параметр  $k$  не прийме нульове значення (у кожному циклі значення  $k$  зменшується на 1). Останнє означає, що максимальний елемент переставлений на місце елемента головної діагоналі. При  $k < 0$  зрушення виконується вліво, а значення  $k$  у кожному циклі збільшується на 1.

У програмі Cont2 передбачено два напрямки виводу даних: на екран дисплея і на принтер. Необхідність використання принтера при черговому запуску програми визначається відповіддю користувача на запит програми.

Процедура ScreenMatrix, що використовується для виводу матриці на екран дисплея, припускає, що для розміщення матриці досить однієї сторінки екрана. Якщо це не так, то ця процедура повинна бути доповнена фрагментом, що забезпечує керування "перелистуванням" сторінок екрана. Приклад такого доповнення для виводу одновимірних масивів наведений у методичних вказівках до лабораторної роботи №1.

Процедура WaitEscape, що міститься в програмі Cont2, використовується для припинення роботи програми доти, поки не буде натиснута клавіша Esc (ця клавіша має код 27).

Змінна  $k$  може приймати значення лише в діапазоні  $-N_{max} .. N_{max}$ . Тому що в програме оголошено  $N_{max} = 30$ . то для розміщення змінної  $k$  досить відвести один байт пам'яті. Цьому відповідають цілочисельні типи `byte` і `shortint`. Оскільки значення  $k$  може бути як додатним, так і цілочисельним, то для цієї змінної в програмі визначений тип `shortint`.

Необхідність використання принтера визначає в програмі змінна `IndPrinter`, рівна `true` при позитивній відповіді на запит програми. Символ `ch`, одержуваний за допомогою функції `ReadKey` при натисканні якої-небудь клавіші, порівнюється зі значеннями символів «Д», «д», «L» і «l». Фактично ці значення генеруються однієї й тією же клавішею (російський й англійський алфавіти, верхній і нижній регістри), що більш ефективно в порівнянні з логічним виразом

```
If (ch='Д') or (ch='д') or (ch='L') or (ch='l') then ...
```

```
Program Cont2;
Uses Crt,Printer;
Const Nmax = 30;
Type
    Matrix = array[1..Nmax,1..Nmax] of real;
Var
    i,j,           { параметри циклу }
    n,             { розмір матриці }
    jmax : byte;  { позиція макс. елемента Amax у рядку }
    k : shortint; { різниця між позиціями Amax }
                { і елемента a[i,i] }
    IndPrinter     { індикатор використання принтера }
                : boolean;
    ch : char;     { символ натиснутої клавіші }
    Amax,          { максимальний елемент у рядку }
    Buf,           { буферна змінна }
    Trace : real;  { слід матриці }
    A : Matrix;   { оброблювана матриця }
    F : text;     { вихідний файл }
{ ----- }
Procedure WaitEscape;
{ Призупинення програми до натискання клавіші Esc }
Var ch : char;
Begin
    Repeat
        ch:=ReadKey;
    Until ord(ch)=27;
End { WaitEscape };
{ ----- }
Procedure ReadMatrix;
{ Читання матриці з текстового файлу }
Var i,j : byte;
Begin
    Reset(F);
    Read(F,n);
    For i:=1 to n do
        For j:=1 to n do
            Read(F,a[i,j]);
    Close(F);
End { ReadMatrix };

```

```

{ ----- }
Procedure ScreenMatrix;
{ Вивід матриці на екран дисплея }
Var i,j,k : byte;
Begin
  For i:=1 to n do
    Begin
      k:=0;
      For j:=1 to n do
        Begin
          Inc(k);
          If k<5 then
            Write(a[i,j]:8:2,' ')
          Else
            Begin
              k:=0;
              Writeln(a[i,j]:8:2)
            End
          End;
        If k>0 then Writeln;
      End;
    End { ScreenMatrix };
{ ----- }
Procedure PrinterMatrix;
{ Вивід матриці на принтер }
Var i,j,k : byte;
Begin
  For i:=1 to n do
    Begin
      k:=0;
      For j:=1 to n do
        Begin
          Inc(k);
          If k<5 then
            Write(Lst,a[i,j]:8:2,' ')
          Else
            Begin
              k:=0;
              Writeln(Lst,a[i,j]:8:2)
            End
          End;
        If k>0 then Writeln(Lst);
      End;
    End { PrinterMatrix };
{ ----- }
Procedure TraceMatrix;
{ Обчислення сліду матриці }
Var i : byte;
Begin
  Trace:=0;
  For i:=1 to n do
    Trace:=Trace+a[i,i];
End { TraceMatrix };
{ ----- }

```



## Begin

```
{ Установлення відповідності між внутрішнім }
{ і зовнішнім файлами }
Assign(F, 'Matrix.dat');

{ Запит про використання принтера }
ClrScr;
Writeln(' Чи буде використовуватися принтер (Так,Hi) ?');
ch:=ReadKey;
If ch in ['Д','д','L','l'] then
    IndPrinter:=true
Else
    IndPrinter:=false;

{ Уведення і друк вхідних даних }
ReadMatrix;
Writeln('          Вхідна матриця   n = ',n);
ScreenMatrix;
If IndPrinter then
    Begin
        Writeln(Lst, '          Вхідна матриця   n = ',n);
        PrinterMatrix;
    End;

{ Обчислення сліду вхідної матриці }
TraceMatrix;
Writeln('Слід вхідної матриці Trace = ',Trace:8:2);
If IndPrinter then
    Writeln(Lst, 'Слід вхідної матриці Trace = ',Trace:8:2);
WaitEscape;

{ Перетворення вхідної матриці }
ClrScr;
For i:=1 to n do
    Begin
        Amax:=a[i,1]; jmax:=1;
        For j:=2 to n do
            If a[i,j]>Amax then
                Begin
                    Amax:=a[i,j]; jmax:=j
                End;
        k:=i-jmax;
        If k>0 then          { циклічне зрушення }
            While k>0 do   { рядка вправо }
                Begin
                    Buf:=a[i,n];
                    For j:=n downto 2 do
                        a[i,j]:=a[i,j-1];
                    a[i,1]:=Buf;
                    Dec(k);
                End
    Else
```

```

If k<0 then                                { циклічне зрушення }
  While k<0 do                               { рядка вліво }
    Begin
      Buf:=a[i,1];
      For j:=1 to n-1 do
        a[i,j]:=a[i,j+1];
      a[i,n]:=Buf;
      Inc(k);
    End;
  End;
Writeln('                                Перетворена матриця');
ScreenMatrix;
If IndPrinter then
  Begin
    Writeln(Lst);
    Writeln(Lst, 'Перетворена матриця');
    PrinterMatrix;
  End;

{ Обчислення сліду перетвореної матриці }
TraceMatrix;
Writeln('Слід перетвореної матриці Trace = ',Trace:8:2);
If IndPrinter then
  Writeln(Lst, 'Слід перетвореної матриці '+
    'Trace = ',Trace:8:2);
WaitEscape;

End.

```

Результати роботи програми:

| Вхідна матриця n = 6 |       |       |       |       |       |
|----------------------|-------|-------|-------|-------|-------|
| 1.00                 | 10.00 | 0.00  | 3.00  | 13.00 | 23.00 |
| 5.00                 | 55.00 | 15.00 | 7.00  | 17.00 | 0.00  |
| 9.00                 | 19.00 | 0.00  | 81.00 | 88.00 | 18.00 |
| 6.00                 | 16.00 | 66.00 | 33.00 | 21.00 | 12.00 |
| 99.00                | 15.00 | 44.00 | 0.00  | 13.00 | 11.00 |
| 10.00                | 11.00 | 12.00 | 17.00 | 23.00 | 14.00 |

Слід вхідної матриці Trace = 116.00

Перетворена матриця

|       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|
| 23.00 | 1.00  | 10.00 | 0.00  | 3.00  | 13.00 |
| 5.00  | 55.00 | 15.00 | 7.00  | 17.00 | 0.00  |
| 0.00  | 81.00 | 88.00 | 18.00 | 9.00  | 19.00 |
| 12.00 | 6.00  | 16.00 | 66.00 | 33.00 | 21.00 |
| 44.00 | 0.00  | 13.00 | 11.00 | 99.00 | 15.00 |
| 14.00 | 10.00 | 11.00 | 12.00 | 17.00 | 23.00 |

Слід перетвореної матриці Trace = 354.00

**ВАРІАНТИ ЗАВДАНЬ ДО КОНТРОЛЬНОЇ РОБОТИ № 1**  
**"ОБРОБКА МАТРИЦЬ"**

---

1. Для прямокутної матриці  $A(m \times n)$  визначити середнє арифметичне значення  $S$  її елементів, середнє квадратичне відхилення  $G$ , а також відсоток елементів матриці, розташованих в інтервалі  $[S - G, S + G]$ .

$$S = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n a_{i,j}; \quad G = \sqrt{\frac{1}{mn-1} \sum_{i=1}^m \sum_{j=1}^n (a_{i,j} - S)^2}$$

---

2. Задано квадратну матрицю  $A(n \times n)$ . Переставити у зворотному порядку елементи тих стовпців матриці, які розташовані нижче її головної діагоналі.

*Примітка.* Цілком зрозуміло, що останній стовпець матриці обробці не підлягає, оскільки в цьому стовпці під головною діагоналлю немає жодного елемента. У передостанньому стовпці під головною діагоналлю перебуває лише один елемент, тому нема рації обмінювати його сам із собою.

---

3. У кожному рядку прямокутної матриці  $A(m \times n)$  від'ємним елементам привласнити нульове значення, після чого перенести всі додатні елементи на початок рядка в порядку їх початкового відносного розташування.

---

4. Кожен рядок прямокутної матриці  $A(m \times n)$  замінити сумою нижче розташованих рядків, включаючи даний рядок. Визначити значення й місце розташування максимального елемента матриці до й після її перетворення.

---

5. Для кожного стовпця прямокутної цілочисельної матриці  $A(m \times n)$  підрахувати суму елементів, які входять до складу цього стовпця і визначити, чи є стовпці з однакою сумою. Підрахувати кількість пар таких стовпців.

*Примітка.* Позначимо кількість пар стовпців з однакою сумою через  $p$ . Легко показати, що в загальному випадку може бути  $p > n$ .

Нехай  $n = 4$  і всі суми однакові. Тоді одержимо пари

$$1 - 2, 1 - 3, 1 - 4, 2 - 3, 2 - 4, 3 - 4.$$

Отже, у цьому випадку  $p = 6 > n$ .

---

6. У кожному рядку й у кожному стовпці квадратної матриці  $A(n \times n)$  міститься строго по одному нульовому елементі. Перестановкою рядків домогтися того, щоб всі нулі були розташовані на головній діагоналі матриці. Буферний масив не використовувати.

---

7. Виконати поточне згладжування кожного рядка прямокутної матриці  $A(m \times n)$  і визначити, як змінилося максимальне відхилення  $dS_{max}$  елементів матриці від її середнього арифметичного значення  $S$  до і після згладжування.

*Примітка.* При поточному згладжуванні масиву  $x_1, x_2, \dots, x_n$  кожний  $j$ -ий елемент масиву ( $j = 2, 3, \dots, n-1$ ) замінюється середнім арифметичним значенням елементів з індексами  $j-1, j, j+1$ . Відхилення елемента  $x_i$  від параметра  $S$  дорівнює  $dS = |x_i - S|$ .

---

8. У кожному рядку прямокутної матриці  $A(m \times n)$  перенести максимальний елемент в останню позицію рядка, зрушивши при цьому вліво розташовані після нього елементи. Врахувати окремий випадок, коли максимальний елемент уже перебуває в останній позиції рядка.

*Приклад.* Рядок

5 18 21 12 10 24 13 17 8 10

після перетворення буде мати вигляд

5 18 21 12 10 13 17 8 10 24 .

---

9. У прямокутній матриці  $A(m \times n)$  кожен нульовий елемент замінити середнім арифметичним значенням ненульових елементів того рядка, у якій розташований даний елемент. При цьому всі нульові елементи рядка повинні бути замінені тим самим значенням.

---

10. У кожному стовпці квадратної матриці  $A(n \times n)$  знайти максимальний по модулю елемент  $i$ , якщо він не є діагональним, поміняти його місцями з діагональним елементом. Підрахувати кількість таких перестановок.

---

11. Дано квадратну матрицю  $A(n \times n)$ . Якщо в її трикутній частині, розташованій вище побічної діагоналі, є нульові елементи, то замінити кожний з них мінімальним, але відмінним від нуля значенням елементів стовпця, у якому розташований нульовий елемент.

*Примітка.* Нульові елементи замінюються лише вище побічної діагоналі, але при пошуку мінімального елемента розглядається весь стовпець !

*Рекомендація.*

1) Організувати цикл перегляду стовпців по  $j$  від 1 до  $n-1$  (при  $j = n$  вище побічної діагоналі немає жодного елемента).

2) В  $j$ -ому стовпці при перегляді його елементів знайти мінімальне, але не рівне нулю значення  $a_{\min}$ .

3) У тім же  $j$ -ому стовпці переглядати лише елементи  $a_{i,j}$ , розташовані вище побічної діагоналі. Якщо  $a_{i,j} = 0$ , то привласнити  $a_{i,j} := a_{\min}$ . Тоді кожний з нульових елементів, розташованих вище побічної діагоналі, замінюється тим самим значенням  $a_{\min}$ .

Якщо в окремому випадку всі елементи стовпця нульові, то ніякої заміни не повинно виконуватися.

---

12. У кожному рядку прямокутної матриці  $A(m \times n)$  підрахувати кількість  $k$  ізольованих додатних елементів, тобто елементів, оточених ліворуч і праворуч хоча б одним недодатним елементом. Для першого елемента суміжними є другий і останній елементи, для останнього – перший і передостанній. Визначити номери рядків, які мають максимальну  $k_{\max}$  і мінімальну  $k_{\min}$  кількість таких елементів, після чого обміняти ці рядки місцями. Урахувати окремий випадок, коли  $k_{\max} = k_{\min}$ .

---

13. Для кожного рядка прямокутної цілочисельної матриці  $A(m \times n)$  визначити суму її додатних елементів, після чого згрупувати рядки в порядку зменшення цих сум.

---

14. У прямокутній матриці  $A(m \times n)$  визначити значення й місце розташування двох елементів, які в максимальному й мініимальному ступені відрізняються від середнього значення сусідніх елементів, після чого обміняти ці елементи місцями. Елементи, розташовані на периметрі матриці, не аналізувати.

*Вказівка.* Для елемента з індексами  $(i, j)$ ,  $i = 2..m-1$ ;  $j = 2..n-1$  сусідніми елементами вважати елементи, суміжні з ним по вертикалі й горизонталі.

---

15. При послідовному перегляді прямокутної цілочисельної матриці  $A(m \times n)$  сформувавши з її елементів одновимірний масив  $B(k)$ ,  $k \leq m \cdot n$ , при цьому в масиві  $B$  не повинно бути однакових чисел.

*Рекомендація.* Перенесемо спочатку в масив  $B$  перший елемент матриці  $A$ , тобто встановимо  $b_1 = a_{1,1}$ ,  $k = 1$  ( $k$  – кількість елементів у масиві  $B$ ). Після цього, переглядаючи в циклі

всі елементи матриці  $A$ , визначити, чи є черговий елемент  $a_{i,j}$  у масиві  $B$  і, якщо такий елемент не виявлений, то додати його в масив  $B$ , збільшивши одночасно значення параметра  $k$ .

---

16. Розглядаючи кожен стовпець прямокутної матриці  $A(m \times n)$  як вектор, визначити номери двох векторів-стовпців, довжини яких відповідно максимальна й мінімальна, після чого обміняти ці стовпці місцями.

*Примітка.* Довжина вектора, представленого одновимірним масивом  $X = (x_1, x_2, \dots, x_n)$ , дорівнює

$$L = \sqrt{\sum_{i=1}^n x_i^2}$$

---

17. Визначити, чи є в прямокутній цілочисельній матриці  $A(m \times n)$  рядки, які являють собою арифметичну прогресію, і підрахувати кількість таких рядків.

*Вказівка.* Якщо числова послідовність  $x_1, x_2, \dots, x_n$  є арифметичною прогресією, то в ній виконується відношення

$$2x_i = x_{i-1} + x_{i+1} \quad (i = 2, 3, \dots, n-1)$$

---

18. У прямокутній матриці  $A(m \times n)$  визначити номери рядків, сума модулів елементів яких найбільша й найменша в порівнянні з іншими рядками, після чого обміняти ці рядки місцями.

---

19. Елементами прямокутної матриці  $A(m \times n)$  можуть бути тільки числа  $-1, 0$  і  $1$ . Для кожного стовпця матриці потрібно виконати наступне: якщо сума елементів стовпця не дорівнює нулю, то замінити частину нульових елементів значенням  $+1$  або  $-1$  таким чином, щоб зазначена сума якнайменше відрізнялася від нуля.

---

20. До кожного елемента головної діагоналі цілочисельної квадратної матриці  $A(n \times n)$  додати таке значення  $d$ , щоб сума елементів у відповідному стовпці матриці була нульовою. Сформувати вектор,  $i$ -ий компонент якого дорівнює значенню параметра  $d$ , після чого згрупувати стовпці матриці в порядку зростання компонент цього вектора.

*Примітка.* Параметр  $d$  може бути як додатним, так і від'ємним.

---

21. Для кожного стовпця прямокутної матриці  $A(m \times n)$ , елементами якої є цілі додатні числа, визначити суму елементів, які входять до його складу, і якщо вона непарна, додати одиницю до значення останнього елемента даного стовпця, після чого згрупувати елементи кожного стовпця в порядку зменшення їхніх значень.

---

22. Розглядаючи кожен рядок прямокутної матриці  $A(m \times n)$  як вектор, знайти номери  $i$ -ого і  $k$ -ого векторів-рядків ( $i = 1 \dots m-1$ ;  $k = i+1 \dots m$ ), для яких модуль їхнього скалярного добутку має максимальне значення, після чого обміняти ці рядки місцями.

*Примітка.* Скалярний добуток двох векторів, які описуються як одновимірні масиви  $X = (x_1, x_2, \dots, x_n)$  й  $Y = (y_1, y_2, \dots, y_n)$ , визначається по формулі

$$S = \sum_{i=1}^n x_i y_i$$

---

23. У прямокутній дійсній матриці  $A(m \times n)$  кожен нульовий елемент замінити середнім арифметичним значенням ненульових елементів того стовпця, у якому розташований цей елемент. При цьому до значення замінюваного елемента додавати одиницю. Перегляд стовпця виконувати зверху вниз, при заміні чергового нульового елемента враховувати значення замінених до цього моменту нульових елементів. Визначити кількість зроблених замінів.

---

24. Головна й побічна діагоналі розділяють елементи квадратної матриці  $A(n \times n)$  на чотири сектори, які умовно можна назвати лівий, правий, верхній і нижній. Вважаючи, що елементи зазначених діагоналей не входять до складу секторів, знайти в лівому секторі мінімальний по модулю, а в правому - максимальний по модулю елементи, після чого обміняти ці елементи місцями.

---

25. У прямокутній матриці  $A(m \times n)$  розглянути квадратні підматриці розмірністю 1, 2, 3, ...,  $n$ , причому для всіх підматриць лівим верхнім елементом є елемент початкової матриці з індексами (1,1). Визначити номери двох підматриць, середнє арифметичне елементів яких має відповідно найбільше й найменше значення.

---

26. У прямокутній матриці  $A(m \times n)$  знайти номери стовпців, які містять відповідно максимальну й мінімальну кількість від'ємних елементів, після чого обміняти їх місцями. Врахувати можливі окремі випадки (у матриці немає від'ємних елементів; лише один стовпець

матриці містить такі елементи; лише два стовпці мають від'ємні елементи, але їх кількість однакова).

---

27. У прямокутній матриці  $A(m \times n)$  визначити середнє арифметичне значення  $S$  і дисперсію  $D$  окремо для додатних і від'ємних елементів даної матриці (нульові елементи при обчисленні  $S$  й  $D$  не враховувати).

*Примітка.*

В цілому для матриці

$$S = \frac{1}{mn} \sum_{i=1}^m \sum_{j=1}^n a_{i,j}; \quad D = \frac{1}{mn-1} \sum_{i=1}^m \sum_{j=1}^n (a_{i,j} - S)^2$$

Якщо в матриці немає додатних елементів, то прийняти  $S = 0, D = 0$ ; якщо в матриці тільки один такий елемент зі значенням  $b$ , те прийняти  $S = b, D = 0$ . Аналогічні дії виконати щодо від'ємних елементів.

---

28. У прямокутній матриці  $A(m \times n)$  визначити кількість рядків, елементи яких повністю впорядковані по зростанню, після чого перенести ці рядки в початкову частину матриці, зберігши їх первісне відносне розташування.

*Примітка.* Елементи масиву  $x_1, x_2, \dots, x_n$  впорядковані по зростанню, якщо  $x_1 \leq x_2 \leq x_3 \leq \dots \leq x_n$ .

*Вказівка.* При перегляді елементів рядка у випадку виявлення пари елементів, які порушують її впорядкованість, тобто має місце  $a_{i,j} > a_{i,j+1}$  ( $j = 1..n-1$ ), подальший аналіз елементів рядка припинити.

---

29. У кожному стовпці прямокутної матриці  $A(m \times n)$  є принаймні один нульовий елемент. Потрібно замінити в стовпці останній нульовий елемент таким значенням, щоб сума елементів стовпця рівнялася нулю. Сформувати вектор,  $i$ -ий компонент якого дорівнює різниці між сумою елементів  $i$ -ого рядка (не стовпця!) до перетворення матриці й сумою елементів цього ж рядка після її перетворення.

---

30. У квадратній матриці  $A(n \times n)$  елементи рядків, які знаходяться вище побічної діагоналі, розташувати в порядку зменшення їх значень, а елементи рядків нижче цієї діагоналі переставити у зворотному порядку.

---



31. У квадратній матриці  $A(n \times n)$  визначити суму додатних елементів, розташованих над головною діагоналлю, і суму від'ємних елементів, розташованих під головною діагоналлю. Якщо перша з них перевищує по модулю другу, то транспонувати матрицю.

---

32. Для прямокутної матриці  $A(m \times n)$  знайти мінімальний з додатних елементів і максимальний з від'ємних елементів, після чого обміняти їх місцями. Нульові елементи не враховувати.

*Вказівка.* Врахувати окремий випадок, коли в матриці немає додатних або від'ємних елементів.

---

33. У прямокутній матриці  $A(m \times n)$  визначити кількість стовпців, які цілком складаються з додатних елементів.

*Вказівка.* Послідовний перегляд елементів стовпця організувати таким чином, щоб при виявленні першого недодатного елемента інші елементи стовпця не перевірялися.

---

34. Частина елементів прямокутної матриці  $A(m \times n)$  має нульове значення. Замінити кожен такий елемент сумою суміжних з ним елементів.

*Примітка.* Кутові елементи матриці мають два суміжних елементи; елементи, розташовані на периметрі матриці, але не кутові, мають три суміжних елементи; інші елементи матриці мають чотири суміжних елементи.

*Вказівка.* Для формування перетвореної матриці в цьому випадку необхідно використати буферний масив (додаткову матрицю).

Нехай .фрагмент матриці має вигляд

|    |    |    |   |
|----|----|----|---|
| 1  | 10 | 3  | 7 |
| -4 | 0  | 0  | 6 |
| 5  | 7  | -8 | 3 |

Якщо перетворення виконується в цій же матриці, то одержимо

|    |    |    |   |
|----|----|----|---|
| 1  | 10 | 3  | 7 |
| -4 | 13 | 14 | 6 |
| 5  | 7  | -8 | 3 |

У цьому випадку при обчисленні нового значення для другого нульового елемента використовується вже обчислене значення першого нульового елемента, що невірно. Тут повинне бути

|    |    |   |   |
|----|----|---|---|
| 1  | 10 | 3 | 7 |
| -4 | 13 | 1 | 6 |

Природно, після формування перетвореної матриці в буферному масиві її елементи потрібно переписати у вхідну матрицю.

35. Розглядаючи елементи рядка прямокутної дійсної матриці  $A(m \times n)$  як координати точки в  $n$ -вимірному просторі, визначити номери точок, відстань між якими максимальна, після чого обміняти їх місцями.

*Примітка.* Якщо дві  $n$ -вимірні точки задані двома одновимірними масивами  $X = (x_1, x_2, \dots, x_n)$  й  $Y = (y_1, y_2, \dots, y_n)$ , то відстань між ними визначається по формулі

$$L = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

36. Дано квадратну цілочисельну матрицю  $A(n \times n)$ . Якщо в її трикутній частині, розташованій нижче головної діагоналі, є непарні елементи, то замінити кожний з них середнім арифметичним значенням  $S$  парних елементів усього стовпця, у якому розташований непарний елемент. Середнє арифметичне значення повинне бути округлене до найближчого цілого парного значення. Якщо в стовпці під головною діагоналлю міститься кілька непарних елементів, то кожний з них повинен бути замінений тим самим значенням.

*Примітка.* Округлення значення  $S$  до найближчого парного значення може бути виконане в такий спосіб:

$$S := \text{trunc}(S)$$

$$a_{i,j} = \begin{cases} S + 1, & \text{якщо } \text{odd}(S) = \text{true} \\ S, & \text{якщо } \text{odd}(S) = \text{false} \end{cases}$$

*Приклад.*

| S   | trunc(S) | $a_{i,j}$ |
|-----|----------|-----------|
| 3.8 | 3        | 4         |
| 3.2 | 3        | 4         |
| 2.8 | 2        | 2         |
| 2.2 | 2        | 2         |

37. До складу прямокутної матриці  $A(m \times n)$  входять цілі додатні числа. Потрібно в кожному рядку матриці визначити суму елементів, які є подвоєними непарними числами, після чого згрупувати рядки в порядку зменшення цих сум.

*Примітка.* Визначити, чи є елемент  $a_{i,j}$  подвоєним непарним числом, можна в такий спосіб:

$$b := a_{i,j}; \quad d := b \operatorname{div} 2$$

Якщо  $\operatorname{odd}(b) = \text{false}$  і  $\operatorname{odd}(d) = \text{true}$ , то елемент  $a_{i,j}$  є подвоєним непарним числом.

---

38. Для прямокутної матриці  $A(m \times n)$ , елементами якої є цілі додатні числа, визначити середнє арифметичне значення  $S$  і середнє квадратичне відхилення  $G$  окремо для парних і непарних елементів. Врахувати, що змінні  $S$  і  $G$  по своїй природі є дійсними числами.

*Примітка.* Див. п.1 і п.27.

---

39. Для прямокутної матриці  $A(m \times n)$  сформувати цілочисельний вектор  $B = (b_1, b_2, \dots, b_m)$ . Елементу  $b_i$  привласнити значення 1, якщо  $i$ -ий рядок має не менше двох підряд розташованих нульових елементів, у протилежному випадку  $b_i = 0$ . Рядки, для яких визначено значення  $b_i = 1$ , перенести на початок матриці, зберігши їх початковий відносний порядок.

*Вказівка.* Якщо при аналізі рядка виявлені два суміжних нульових елементи, подальший перегляд рядка припинити.

---

40. Для заданої цілочисельної квадратної матриці  $A(n \times n)$  перевірити, чи має місце збіг  $k$ -ого рядка і  $k$ -го стовпця ( $k = 1..n$ ). Надрукувати номери співпадаючих рядків і стовпців.

---

41. Задана цілочисельна квадратна матриця  $A(n \times n)$ . Згрупувати елементи кожного рядка, розташовані вище головної діагоналі, у порядку зменшення їхніх абсолютних значень.

---

42. Якщо максимальний елемент  $i$ -ого рядка прямокутної матриці  $A(m \times n)$ ,  $i = 1..m$  більше суми інших елементів даного рядка, а модуль мінімального елемента менше суми інших елементів рядка, то замінити максимальний і мінімальний елементи напівсумою їхніх значень, у протилежному випадку рядок матриці залишити без змін. Підрахувати кількість рядків матриці, у яких зроблена зазначена заміна.

---

43. Нульові елементи кожного стовпця прямокутної матриці  $A(m \times n)$  перемістити на початок цього ж стовпця, зберігши без зміни відносну послідовність інших елементів стовпця.

---

44. У кожному рядку цілочисельної прямокутної матриці  $A(m \times n)$  знайти максимальний по модулю елемент і замінити його середнім арифметичним значенням  $S$  інших елементів даного рядка. Параметр  $S$  повинен бути округлений до найближчого цілого значення.

---

45. Розглядаючи кожен стовпець прямокутної дійсної матриці  $A(m \times n)$  як вектор, визначити номери  $j$  і  $k$  ( $j = 1 \dots n-1$ ;  $k = j+1 \dots n$ ) векторів-стовпців, кут між якими максимальний, після чого обміняти ці стовпці місцями.

*Примітка.* Косинус кута між двома векторами - це скалярний добуток векторів, розділений на добуток їхніх модулів (модуль вектора - це його довжина). Якщо два вектори описані одновимірними масивами  $X = (x_1, x_2, \dots, x_n)$  і  $Y = (y_1, y_2, \dots, y_n)$ , то тоді маємо

$$\cos \alpha = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}$$

Кут  $\alpha$  максимальний, якщо модуль значення  $\cos \alpha$  мінімальний.

---

46. У кожному рядку прямокутної матриці  $A(m \times n)$  визначити середнє арифметичне значення  $S$  його елементів, після чого елементам  $a_{i,j} < S$  привласнити нульове значення. Згрупувати рядки матриці в порядку зменшення параметра  $S$ .

---

47. Задана цілочисельна прямокутна матриця  $A(m \times n)$ . Якщо сума елементів її  $i$ -го рядка ( $i = 1..m$ ) непарна, то додати одиницю до значення останнього елемента  $a_{i,n}$ , після чого згрупувати елементи цього рядка в порядку зменшення їх абсолютних значень.

---

48. У кожному рядку прямокутної матриці  $A(m \times n)$  вилучити максимальний елемент, зрушивши на одну позицію вліво розташовані після нього елементи. Останньому елементу рядка привласнити нульове значення. Визначити, як змінилося при цьому середнє арифметичне значення елементів матриці.

---

49. Задано верхню трикутну матрицю  $A(n \times n)$  ( $a_{i,j} = 0$  для всіх  $i > j$ ;  $i, j = 1..n$ ) з дійсними елементами. У кожному рядку цієї матриці, крім першого, замінити нульові елементи такими рівними один одному значеннями, щоб сума елементів даного рядка стала рівною нулю.

*Примітка.* Елементами верхньої трикутної матриці вважаються елементи, розташовані над головною діагоналлю, і елементи головної діагоналі. Елементи під головною діагоналлю при цьому повинні мати нульове значення.

---

50. У кожному  $i$ -му рядку прямокутної матриці  $A(m \times n)$  знайти кількість  $k$  елементів  $a_{i,j}$ , які перевищують напівсуму елементів  $a_{i,j-1}$  і  $a_{i,j+1}$ , суміжних з елементом  $a_{i,j}$ , після чого згрупувати рядки матриці в порядку зростання параметра  $k$ . Перший і останній елементи рядка не розглядати.

---

## КОМАНДИ РЕДАКТОРА СИСТЕМИ ТУРБО ПАСКАЛЬ

При перегляді й редагуванні вмісту текстового файлу, зокрема тексту Паскаль-програми, використовуються відповідні команди редактора, які задаються шляхом натискання однієї, двох або трьох клавішею. Якщо в команді зазначено дві клавіші, то при натиснутій першій клавіші виробляється натискання другої клавіші (наприклад, *Ctrl+PgUp*); якщо зазначені три клавіші, то при натиснутій першій клавіші виробляється натискання другої, а після її відпускання - натискання третьої клавіші (наприклад, *Ctrl+K, B*).

Команди редактора діляться на чотири групи:

- команди переміщення курсору
- команди вставки й видалення;
- команди роботи із блоками;
- додаткові команди.

## 1. Команди переміщення курсору

← (*Left*) - зрушення курсору на один символ уліво;

→ (*Right*) - зрушення курсору на один символ вправо;

↑ (*Up*) - зрушення курсору на один рядок нагору;

↓ (*Down*) - зрушення курсору на один рядок униз;

*PgUp* (*Page Up*) - сторінка нагору;

*PgDn* (*Page Down*) - сторінка вниз;

*Home* - переміщення курсору в першу позицію рядка;

*End* - переміщення курсору в кінець рядка;

*Ctrl+Home* - переміщення курсору на початок сторінки;

*Ctrl+End* - переміщення курсору в кінець сторінки;

*Ctrl+PgUp* - переміщення курсору на початок файлу;

*Ctrl+PgDn* - переміщення курсору в кінець файлу;

*Ctrl+W* - скролінг нагору;

*Ctrl+Z* - скролінг униз;

*Tab* - табулированне переміщення курсору. Якщо попередній рядок праворуч від курсору не порожня, то при кожнім натисканні клавіші *Tab* курсор переміщається на початок найближчого слова на попередньому рядку (слово - це група символів, відмінних від пробілу); у протилежному випадку курсор переміщається на 8 позицій вправо. Клавішу *Tab* звичайно використовують при формуванні нового текстового файлу.

Примітка. Скролінг (прокручування) - це переміщення екранного тексту нагору або вниз.

## 2. Команди вставки й видалення

При натисканні будь-якої алфавітно-цифрової або знакової клавіші виробляється вставка або заміщення символу в рядку над курсором. Зміна режиму вставки-заміщення відбувається при натисканні клавіші *Ins*. У першому випадку на екран виводиться вузький курсор, у другому - широкий, на всю висоту символу.

*Del* - видалення символу над курсором;

*BackSpace* - видалення символу ліворуч від курсору;

*Enter* - перехід на наступний рядок; якщо курсор перебуває не наприкінці поточного рядка, то на новий рядок переноситься права частина поточного рядка, починаючи з позиції курсору;

*Ctrl+N* - вставка порожнього рядка над курсором;

*Ctrl+Y* - видалення рядка над курсором;

*Ctrl+Q,Y* - видалення правої частини поточного рядка, починаючи з позиції курсору.

### 3. Команди роботи із блоками

Блоком вважається виділена підсвічуванням частина рядка файлу, однієї або декількох послідовних рядків файлу, в окремому випадку весь файл.

*Ctrl+K, B* - маркування початку блоку;

*Ctrl+K, K* - маркування кінця блоку;

*Ctrl+K, H* - демаркировка блоку;

*Ctrl+K, C* - копіювання підсвіченого блоку (починаючи з рядка, певної положенням курсору поза блоком);

*Ctrl+K, V* - переміщення блоку;

*Ctrl+K, I* - зрушення блоку на одну позицію вправо;

*Ctrl+K, U* - зрушення блоку на одну позицію вліво;

*Ctrl+K, Y* - видалення блоку;

*Ctrl+K, P* - печатка блоку;

*Ctrl+K, R* - приєднання файлу;

*Ctrl+K, W* - запис блоку у файл.

По команді *Ctrl+K, R* до файлу, викликаному на згадку редактора, приєднується файл, ім'я якого зазначене користувачем по запиті програми.

По команді *Ctrl+K, W* блок, відзначений підсвічуванням, записується у файл, ім'я якого зазначено користувачем по запиті програми.

### 4. Додаткові команди

*Ctrl+Q,F* - шукати за зразком;

*Ctrl+L* - продовжити пошук;

*Ctrl+Q,A* - шукати за зразком і замінити;

*Ctrl+U* - припинити подальше виконання пошуку або пошуку й заміни;

*Ctrl+Q,L* - відновити зіпсований рядок.

Інформацію із всіх команд редактора можна одержати при натисканні клавіші F1.