

Частина 3. Лабораторний практикум

Загальні вимоги до лабораторних робіт

1. Структура Pascal-програми. Введення-виведення
2. Лінійний алгоритм. Числові типи даних
3. Лінійний алгоритм. Логічні типи даних
4. Розгалужений алгоритм. Оператор if
5. Розгалужений алгоритм. Оператор case
6. Циклічний алгоритм. Оператор.for
7. Циклічний алгоритм. Оператори while і repeat
8. Структуризація програм. Функції
9. Структуризація програм. Процедури
10. Масиви
11. Обробка символічної інформації
12. Множини
13. Записи
14. Файли
15. Модулі

Лабораторний практикум

Загальні вимоги до лабораторних робіт	Варіанти індивідуальних завдань	Перелік лабораторних робіт	Приклад оформлення звіту
---	---	--	--

Загальні вимоги до лабораторних робіт

Кожна лабораторна робота містить короткі рекомендації, які необхідні для виконання лабораторної роботи. Запропоновані завдання виконуються студентом особисто. Номер індивідуального варіанту завдання повинен відповідати номеру, що визначається таблицею наведеною далі. Допускається об'єднання зусиль студентів у яких співпав номер варіанту завдання. Захист лабораторної роботи здійснюється кожним студентом особисто.

Звіт про лабораторну роботу (див. [приклад](#)) оформлюється на аркушах ф. А4 (без рамок) з полем зліва 20 мм і повинен мати таку структуру:

- Номер і назва лабораторної роботи;
- Мета лабораторної роботи;
- Індивідуальне завдання до лабораторної роботи;
- Загальний алгоритм вирішення задачі;
- Текст програми;
- Копія з екрану результату роботи програми;

- Висновки по роботі;
- Інформація про виконавця роботи і викладача, що її прийняв.

При здачі лабораторних робіт кожен студент повинен знати відповіді на контрольні запитання. Програми повинні оформлюватись охайно і з відповідними коментарями до основних блоків програми.

Відповідальність за збереження програм на жорсткому диску і особистій дискеті несе студент особисто.

Варіанти індивідуальних завдань

№ п/п	Гр. МТ- 1											
	№ лабораторної роботи											
	2	3	4	5	6	7	8	9	10	11	12	13
1	1	2	3	4	5	6	7	8	9	10	11	12
2	2	3	4	5	6	7	8	9	10	11	12	13
3	3	4	5	6	7	8	9	10	11	12	13	14
4	4	5	6	7	8	9	10	11	12	13	14	15
5	5	6	7	8	9	10	11	12	13	14	15	16
6	6	7	8	9	10	11	12	13	14	15	16	17
7	7	8	9	10	11	12	13	14	15	16	17	18
8	8	9	10	11	12	13	14	15	16	17	18	19
9	9	10	11	12	13	14	15	16	17	18	19	20
10	10	11	12	13	14	15	16	17	18	19	20	1
11	11	12	13	14	15	16	17	18	19	20	1	2
12	12	13	14	15	16	17	18	19	20	1	2	3
13	13	14	15	16	17	18	19	20	1	2	3	4
14	14	15	16	17	18	19	20	1	2	3	4	5
15	15	16	17	18	19	20	1	2	3	4	5	6
16	16	17	18	19	20	1	2	3	4	5	6	7
17	17	18	19	20	1	2	3	4	5	6	7	8
18	18	19	20	1	2	3	4	5	6	7	8	9
19	19	20	1	2	3	4	5	6	7	8	9	10
20	20	1	2	3	4	5	6	7	8	9	10	11
21	1	3	5	7	9	11	13	15	17	19	2	4
22	2	4	6	8	10	12	14	16	18	20	1	3
23	3	5	7	9	11	13	15	17	19	2	4	6
24	4	6	8	10	12	14	16	18	20	1	3	5
25	5	7	9	11	13	15	17	19	2	4	6	8
26	6	8	10	12	14	16	18	20	1	3	5	7
27	7	9	11	13	15	17	19	2	4	6	8	10
28	8	10	12	14	16	18	20	1	3	5	7	9
29	9	11	13	15	17	19	2	4	6	8	10	12
30	10	12	14	16	18	20	1	3	5	7	9	11

№ п/п	Гр. МТ- 2											
	№ лабораторної роботи											
	2	3	4	5	6	7	8	9	10	11	12	13
1	20	19	18	17	16	15	14	13	12	11	10	9
2	19	18	17	16	15	14	13	12	11	10	9	8
3	18	17	16	15	14	13	12	11	10	9	8	7
4	17	16	15	14	13	12	11	10	9	8	7	6
5	16	15	14	13	12	11	10	9	8	7	6	5
6	15	14	13	12	11	10	9	8	7	6	5	4
7	14	13	12	11	10	9	8	7	6	5	4	3
8	13	12	11	10	9	8	7	6	5	4	3	2
9	12	11	10	9	8	7	6	5	4	3	2	1
10	11	10	9	8	7	6	5	4	3	2	1	20
11	10	9	8	7	6	5	4	3	2	1	20	19
12	9	8	7	6	5	4	3	2	1	20	19	18
13	8	7	6	5	4	3	2	1	20	19	18	17
14	7	6	5	4	3	2	1	20	19	18	17	16
15	6	5	4	3	2	1	20	19	18	17	16	15
16	5	4	3	2	1	20	19	18	17	16	15	14
17	4	3	2	1	20	19	18	17	16	15	14	13
18	3	2	1	20	19	18	17	16	15	14	13	12
19	2	1	20	19	18	17	16	15	14	13	12	11
20	1	20	19	18	17	16	15	14	13	12	11	10
21	20	18	16	14	12	10	8	6	4	2	19	17
22	19	17	15	13	11	9	7	5	3	1	20	18
23	18	16	14	12	10	8	6	4	2	19	17	15
24	17	15	13	11	9	7	5	3	1	20	18	16
25	16	14	12	10	8	6	4	2	19	17	15	13
26	15	13	11	9	7	5	3	1	20	18	16	14
27	14	12	10	8	6	4	2	19	17	15	13	11
28	13	11	9	7	5	3	1	20	18	16	14	12
29	12	10	8	6	4	2	19	17	15	13	11	9
30	11	9	7	5	3	1	20	18	16	14	12	10

№ п/п	Гр. МТ- 3											
	№ лабораторної роботи											
	2	3	4	5	6	7	8	9	10	11	12	13
1	1	3	5	7	9	11	13	15	17	19	2	4
2	3	5	7	9	11	13	15	17	19	2	4	6
3	5	7	9	11	13	15	17	19	2	4	6	8
4	7	9	11	13	15	17	19	2	4	6	8	10
5	9	11	13	15	17	19	2	4	6	8	10	12
6	11	13	15	17	19	2	4	6	8	10	12	14
7	13	15	17	19	2	4	6	8	10	12	14	16
8	15	17	19	2	4	6	8	10	12	14	16	18
9	17	19	2	4	6	8	10	12	14	16	18	20
10	19	2	4	6	8	10	12	14	16	18	20	1
11	2	4	6	8	10	12	14	16	18	20	1	3
12	4	6	8	10	12	14	16	18	20	1	3	5
13	6	8	10	12	14	16	18	20	1	3	5	7
14	8	10	12	14	16	18	20	1	3	5	7	9
15	10	12	14	16	18	20	1	3	5	7	9	11
16	12	14	16	18	20	1	3	5	7	9	11	13
17	14	16	18	20	1	3	5	7	9	11	13	15
18	16	18	20	1	3	5	7	9	11	13	15	17
19	18	20	1	3	5	7	9	11	13	15	17	19
20	20	1	3	5	7	9	11	13	15	17	19	2
21	1	4	7	10	13	16	19	2	5	8	11	14
22	3	6	9	12	15	18	1	4	7	10	13	16
23	5	8	11	14	17	20	3	6	9	12	15	18
24	7	10	13	16	19	2	5	8	11	14	19	2
25	9	12	15	18	1	4	7	10	13	16	1	4
26	11	14	17	20	3	6	9	12	15	18	3	6
27	13	16	19	2	5	8	11	14	19	2	5	8
28	15	18	1	4	7	10	13	16	1	4	7	10
29	17	20	3	6	9	12	15	18	3	6	9	12
30	19	2	5	8	11	14	19	2	5	8	11	14

Перелік лабораторних робіт

1. [Структура Pascal - програми. Введення-виведення даних](#)
2. [Лінійний алгоритм. Числові типи даних](#)
3. [Лінійний алгоритм. Логічні типи даних](#)
4. [Розгалужений алгоритм. Оператор if](#)
5. [Розгалужений алгоритм. Оператор case](#)
6. [Циклічний алгоритм. Оператор For](#)
7. [Циклічний алгоритм. Оператори While i Repeat ... until](#)
8. [Структуризація програм. Функції](#)
9. [Структуризація програм. Процедури](#)
10. [Масиви](#)
11. [Обробка символічної інформації](#)
12. [Множини](#)
13. [Записи](#)
14. [Файли](#)
15. [Модулі](#)

Приклад оформлення звіту з лабораторних робіт

Лабораторна робота № 4

Розгалужений алгоритм. Оператор IF

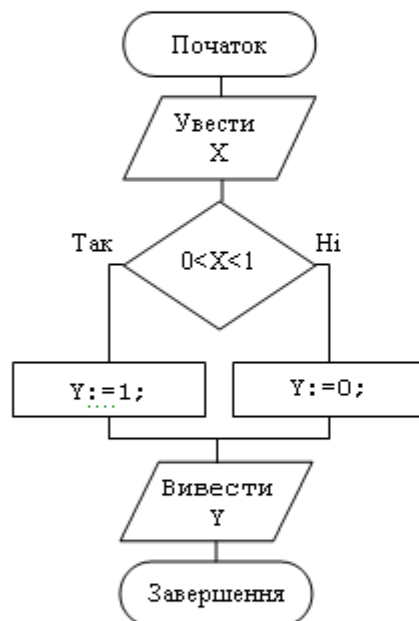
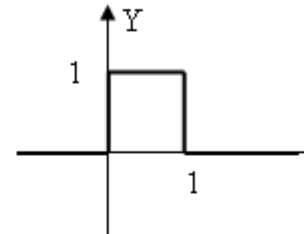
Мета роботи: здобуття практичних навичок створення розгалужених алгоритмів з використанням оператора перевірки умови IF.

Завдання (варіант № 7)

Написати програму для обчислення значення функції, яка задана графічно

Алгоритм вирішення задачі

Рішення цієї задачі реалізується розгалуженим алгоритмом, який зображений нижче.

**Програма:**

```

Program Lab4_v7;
Var
  X, Y: real;
Begin
  Write('X = '); readln(X);
  If (X > 0) and (X < 1)
  then Y := 1
  else Y := 0;
  Writeln('Y = ', Y:8:2);
End.
  
```

Результат роботи:

X = -2
Y = 0.00

X = 0.5
Y = 1.00

Висновки: При виконанні лабораторної роботи я створив програму з розгалуженим алгоритмом роботи і здобув навички роботи з оператором перевірки умови IF.

Виконав: ст. гр. МТ-21
Коваленко В. П.

Перевірив:
доцент Пасічник В. А.

[Загальні вимоги до лабораторних робіт](#)

[Варіанти індивідуальних завдань](#)

[Перелік лабораторних робіт](#)

[Приклад оформлення звіту](#)

Лабораторна робота № 1

Структура Pascal-програми. Введення-виведення даних

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок набирання і редагування текстів програм мовою Pascal, засвоєння використання операторів введення-виведення, набуття досвіду створення найпростіших програм.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 11](#), питання [1 з теми 12](#), уважно розглянути [приклад 12.1](#). При вирішенні завдань цієї лабораторної роботи слід пам'ятати по так:

- Кожна програма складається з двох розділів: розділу описування і розділу операторів;
- Кожна змінна або константа, що використовується програмою повинна бути описаною до початку розділу операторів;
- Описування констант проводиться у розділі, що починається зі слова `const`. Саме описування константи виглядає так:
Ім'яКонстанти = Значення;
тип константи автоматично визначається компілятором;
- Описування змінних проводиться у розділі, що починається зі слова `var`. Саме описування змінної виглядає так:
Ім'яЗмінної : Тип;
- Імена змінних і констант складаються з латинських букв і цифр (першим символом повинна бути буква);
- Рекомендується після визначення констант і змінних коментарями вказувати їхнє призначення у програмі;
- Для введення початкових даних (значень змінних) з клавіатури під час виконання програми використовується оператор `readln (read)`. Відмінність оператора `readln` від оператора `read` полягає у тому, що перший з них після введення чергової порції даних переводить курсор у початок рядка екрану;
- Тип даних, що вводяться під час роботи програми, повинен відповідати типові змінної, якій присвоюється значення за допомогою операторів `readln` або `read`. Коли тип уведеного значення не відповідає типові змінної, програма аварійно завершує свою роботу і на екран видається повідомлення `Error 106: Invalid numeric format` (Невірний числовий формат – при запуску програми із середовища Turbo Pascal) або `Run time error 106` (Помилка виконання програми – при запуску програми із операційної системи);
- Рекомендується для введення даних використовувати зв'язку операторів `write (...); readln (...);`
- Для виведення значень змінних, констант, виразів або текстових коментарів використовуються оператори `writeln (write)`. Відмінність оператора `writeln` від оператора `write` полягає у тому, що перший з них після виведення інформації на екран переводить курсор у початок нового рядка екрану;
- Один оператор `writeln (write)` може вивести на екран декілька значень змінних, констант, виразів або коментарів до них. У такому випадку вони розділяються між собою комами;
- При виведенні на екран рекомендується використовувати формати виведення (особливо для числових даних дійсного типу);

- Не забувайте ставити „;” після кожного оператора Pascal-програми. У випадку його відсутності у потрібному місці, під час компіляції на екран буде виведене повідомлення Error 85: „;” expected. (потребується „;”), а курсор вказує позицію **перед якою** потрібен цей службовий символ.

Варіанти завдань

Створіть Pascal-програму, у якій константами визначте день, місяць і рік свого народження, своє прізвище, ім'я та по батькові. Визначте змінні, у яких будуть зберігатися: курс на якому Ви навчаєтеся, Код групи, середній бал атестату про середню освіту. Виведіть на екран всю інформацію, у вигляді, що рекомендується нижче. Перед виведенням інформації очистіть екран.

Рекомендований вигляд результату роботи програми:

Відомості про студента 1 курсу:

Група : МТ-21с
Прізвище : Коваленко
Ім'я : Василь
По батькові : Петрович
Дата народження: 15 жовтня 1984 року
Середній бал : 4.25

Контрольні запитання

1. З яких розділів складається Pascal-програма, які функції вони виконують?
2. Що таке заголовок Pascal-програми, для чого і чи обов'язково він потрібен?
3. Скільки розділів містить розділ описування Pascal-програми, чи може він бути пустим?
4. Які обмеження існують на порядок і кількість розділів розділу описування Pascal-програми?
5. Наведіть приклади описування констант, типів і змінних Pascal-програми.
6. Поясніть різницю між процедурами `read` і `readln`.
7. Поясніть різницю між процедурами `write` і `writeln`.
8. Як будуть виводитись вирази на екран без явного вказування форматів виведення? Наведіть приклади.
9. Як будуть виводитись вирази, якщо після них через двокрапку вказати ціле число? Наведіть приклади.
10. Як будуть виводитись дійсні вирази, якщо після них через двокрапки вказати два цілих числа? Наведіть приклади.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 2

Лінійний алгоритм. Числові типи даних

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок роботи зі стандартними числовими типами даних і створення програм з лінійним алгоритмом вирішення задачі.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [1 з теми 12](#), уважно розглянути [приклад 12.2](#). При вирішенні завдань цієї лабораторної роботи слід пам'ятати по таке:

- Перед виконанням роботи слід обов'язково з'ясувати за якими формулами слід розраховувати потрібні значення. В деяких варіантах ці залежності задані явно (вар. 10, 17-19), в більшості варіантів залежність і послідовність її формування слід отримати самостійно;
- В програмах із лінійною структурою інструкції виконуються послідовно одна за одною;
- У загальному випадку програми з лінійною структурою складаються з таких блоків: введення даних з клавіатури, обробка інформації, виведення результатів на екран монітора;
- Пам'ятайте про те, що будь-яку залежність або формулу слід записати у вигляді, який є прийнятним для Pascal, тобто таку залежність слід записати у рядок, вказавши пріоритет виконання операцій круглими дужками;
- Якщо задача не вирішується одразу, слід розбити її на простіші етапи;

Варіанти завдань

1. Цілій змінній *Suma* присвоїти суму цифр тризначного цілого числа *K*, значення якого вводиться з клавіатури.
2. Дійсній змінній *Aver* присвоїти середнє значення цифр тризначного цілого числа *K*, значення якого вводиться з клавіатури.
3. Цілій змінній *Des* присвоїти першу цифру з дробової частини додатного дійсного числа *X*, значення якого вводиться з клавіатури (так, якщо $X = 32.597$, $Des=5$).
4. З клавіатури вводиться значення, яке вказує, скільки секунд пройшло від початку доби. Це значення присвоюється змінній *Sec*. Визначити скільки повних годин (*Hour*) і повних хвилин (*Minute*) пройшло до цього моменту (наприклад, якщо $Sec=13255$, тоді $Hour=3$, $Minute=40$, тобто, $=3*3600+40*60+57$).
5. З клавіатури вводиться кількість годин (*Hour*), хвилин (*Minute*) і секунд (*Sec*) поточного часу. Визначити скільки всього секунд пройшло від початку доби.
6. Визначити *Angle* – кут (у градусах) між положенням годинникової стрілки на початку доби та її положенням в *Hour* годин, *Minute* хвилин, *Sec* секунд. Значення *Hour*, *Minute*, *Sec* вводяться з клавіатури.
7. Визначити *Hour* – повну кількість годин і *Minute* – повну кількість хвилин, які пройшли від початку доби до того моменту (в першій половині доби), коли годинникова стрілка повернулася на *Angle* градусів. Значення *Angle* вводиться з клавіатури.
8. Нехай *Day* – ціле число в межах від 1 до 365. Присвоїти цілій змінній *DayWeek* значення 1,2,...,7 в залежності від того, на який день тижня припадає день *Day*. День тижня для 1 січня визначається константою *FirstDay*.
9. Визначити суму, різницю, добуток, частку і середнє арифметичне значення двох цілих чисел *X* і *Y* значення яких вводяться з клавіатури.
10. З клавіатури вводяться два дійсних числа *X* і *Y*. Визначити число *Z*, у відповідності з формулою:

$$z = |(0,866*10^{-14}*X - 1,4123*10^{-12}*Y)| + |(X-Y)*10^{-13}|.$$
11. Змішали *V1* літрів води, що має температуру *T1*, і *V2* літрів води, що має температуру *T2*. Знайти об'єм *Vs* і температуру *Ts* утвореної суміші.
12. З клавіатури вводяться довжина, ширина, висота паралелепіпеда і густина матеріалу, з якого він зроблений. Знайти площу поверхні, об'єм і масу паралелепіпеда.
13. З клавіатури вводяться радіус, висота циліндра і густина матеріалу з якого він зроблений. Знайти площу поверхні, об'єм і масу циліндра.

14. З клавіатури вводяться більший і менший радіуси, висота полого циліндра і густина матеріалу з якого він зроблений. Знайти площу поверхні, об'єм і масу такого циліндра.
15. З клавіатури вводяться радіус кулі і густина матеріалу з якого вона зроблена. Знайти площу поверхні, об'єм і масу такої кулі.
16. З клавіатури вводяться радіус, висота конуса і густина матеріалу з якого він зроблений. Знайти площу поверхні, об'єм і масу конуса.
17. Значення змінних X і Y ($X < Y$) вводяться з клавіатури. Решта значень присвоюється як константи. Розрахувати значення виразу за такою формулою:
$$Z = \frac{X+Y}{a_1} \cdot \frac{a_2}{X-Y}$$
18. Значення змінних M_1, M_2, R вводяться з клавіатури. Розрахувати значення виразу за такою формулою:
$$F = 6,673 \cdot 10^{-8} \frac{m_1 \cdot m_2}{r^2}$$
19. Значення змінних X і Y вводяться з клавіатури. Розрахувати значення виразу за такою формулою:
$$Z = \left(X + \frac{1}{2}\right) \cdot \left(Y + \frac{7}{10}\right) - \frac{3}{4}$$
20. Значення змінних A, B, C є відповідними коефіцієнтами квадратного рівняння виду $Ax^2+Bx+C=0$ і вводяться з клавіатури. Розрахувати значення дискримінанту квадратного рівняння.

Контрольні запитання

1. Наведіть приклади цілочислових змінних і констант. Який об'єм пам'яті вони займають?
2. Наведіть приклади дійсних змінних і констант. Який об'єм пам'яті вони займають?
3. Що таке вираз, операнд, операція? Поясніть на прикладі.
4. Які типи операцій ви знаєте? Наведіть приклади.
5. Наведіть приклади арифметичних виразів і операцій.
Який тип даних ви отримаєте при діленні двох змінних цілого типу?
6. Поясніть дію операцій `div` і `mod`.
7. Поясніть дію функцій `abs`, `sqrt`, `sqr`.
8. Поясніть дію функцій `trunc`, `frac`, `round`, `int`.
9. Поясніть дію функцій `inc`, `dec`.
10. Прикладом проілюструйте пріоритет виконання операцій.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 3

Лінійний алгоритм. Логічні типи даних

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок роботи зі стандартними логічними типами даних і створення програм з лінійним алгоритмом вирішення задачі.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [1 з теми 12](#), уважно розглянути [приклад 12.3](#).

При вирішенні завдань цієї лабораторної роботи слід пам'ятати по таке:

- Перш за все слід усвідомити, що логічну інформацію можна обробляти так само, як і числову, тільки замість арифметичних операцій слід застосовувати логічні операції, до яких відносяться: AND – логічне І; OR – логічне АБО; XOR – логічне виключне АБО; NOT – логічне НІ;
- Результатом логічного виразу завжди буде логічне дане, представлене значеннями `true` або `false`;
- Якщо до складу логічного виразу входять цілі або дійсні числа, то вони повинні порівнюватися між собою за допомогою операцій відношення, таких як: `>` більше; `<` менше; `>=` не менше; `<=` не більше; `<>` не дорівнює; `=` дорівнює. Для цілих чисел, додатково можна застосовувати операцію приналежності `IN`;
- При аналізі дійсних чисел не рекомендується вживати операції чіткої рівності (`=`) або нерівності (`<>`). Замість цього краще порівнювати різницю між значеннями з похибкою, яку ви припускаєте, наприклад, замість `X<>Y`, записати так `(X-Y)>0.0001`;
- Пам'ятайте, що операції відношення у порівнянні з логічними операціями мають нижчий пріоритет, тому найчастіше для правильного запису логічного виразу їх оточують круглими дужками;
- Якщо до складу логічного виразу входять символні або рядкові дані, вони повинні порівнюватися між собою, причому саме порівнювання відбувається по символах у відповідності до коду символу у кодовій таблиці комп'ютера;
- Створюючи логічний вираз слід передбачити усі можливі варіанти, а не обмежуватися одним знайденим рішенням;
- При вирішенні завдань на шаховій дошці найчастіше слід аналізувати різницю положення фігур, для покращення сприйняття – намалюйте шахову дошку і проставте необхідні фігури;
- При вирішенні задач на попадання точки у заштриховану зону, слід аналітично представити геометричні елементи з яких така зона складається та порівняти положення точки з цими елементами;
- Для скорочення логічних виразів слід використовувати властивості симетричності відносно однієї або двох осей координат;

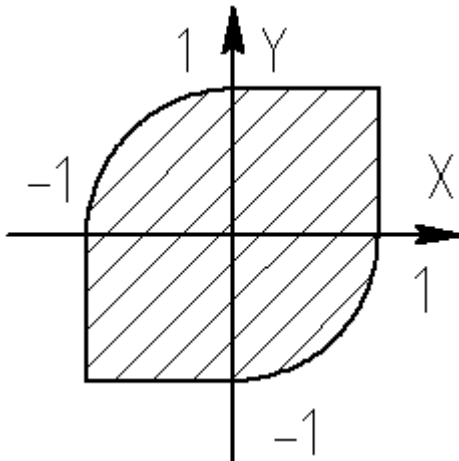
Варіанти завдань

Примітка. Зверніть увагу на те, що усі завдання цієї лабораторної роботи повинні бути вирішені без використання оператора `IF`. Результатом завжди буде логічне значення `true` або `false`.

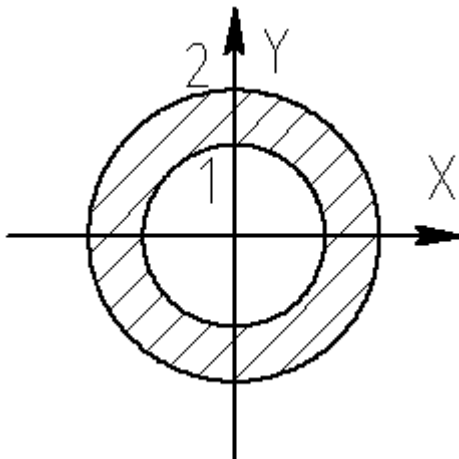
1. Логічній змінній `Nachalo` присвоїти значення `true`, якщо початок системи координат попадає всередину прямокутника, координати вершин якого визначаються дійсними змінними `(X1, Y1)`, `(X2, Y2)`. Значення `X1, Y1, X2, Y2` вводяться з клавіатури. У програмі слід передбачити різні варіанти розташування точок 1 і 2 на координатній площині.
2. Логічній змінній `Triangle` присвоїти значення `true`, якщо існує трикутник, координати вершин якого визначаються дійсними змінними `(X1, Y1)`, `(X2, Y2)`, `(X3, Y3)`. Значення `X1, Y1, X2, Y2, X3, Y3` вводяться з клавіатури.
3. З'ясувати, чи можна прямокутник зі сторонами `A, B` розмістити всередині прямокутника зі сторонами `C, D` так, щоб кожна зі сторін одного прямокутника була паралельна або перпендикулярна до кожної сторони іншого прямокутника. Результат присвоїти логічній змінній `Rozmist`.
4. З'ясувати, чи пройде цеглина з ребрами `L, B, H` у прямокутний отвір з сторонами `X, Y`. Результат присвоїти логічній змінній `Se`.
5. Логічній змінній `Visokos` присвоїти логічне значення `true` якщо рік є високосним.

Порядковий номер року визначається змінною Rik . Рік є високосним, якщо його номер ділиться на 4 без остачі, однак з років, що діляться без остачі на 100, високосними є лише ті, які діляться без остачі на 400, тобто 1700, 1800 – не високосні роки, 2000 – високосний рік.

6. Логічній змінній $Pole$ присвоїти значення `true`, якщо поля $(Hor1, Ver1)$ і $(Hor2, Ver2)$ шахової дошки мають однаковий колір ($Hor1, Ver1, Hor2, Ver2$ – цілі від 1 до 8).
7. Логічній змінній Voj присвоїти значення `true`, якщо ферзь, розташований на полі $(Hor1, Ver1)$, б'є поле $(Hor2, Ver2)$ шахової дошки ($Hor1, Ver1, Hor2, Ver2$ – цілі від 1 до 8).
8. Логічній змінній Voj присвоїти значення `true`, якщо кінь, розташований на полі $(Hor1, Ver1)$, б'є поле $(Hor2, Ver2)$ шахової дошки ($Hor1, Ver1, Hor2, Ver2$ – цілі від 1 до 8).
9. З'ясувати, чи можна за один хід перемістити туру з поля $(Hor1, Ver1)$ на поле $(Hor2, Ver2)$. Вважати, що тура знаходиться на дошці одна. Результат присвоїти логічній змінній $Ladya$.
10. З'ясувати, чи можна за один хід перемістити слона з поля $(Hor1, Ver1)$ на поле $(Hor2, Ver2)$. Вважати, що слон знаходиться на дошці один. Результат присвоїти логічній змінній $Slon$.
11. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких уводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній $Target$.

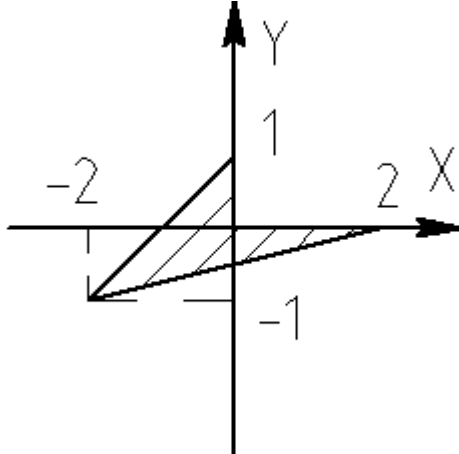


12. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких уводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній $Target$.

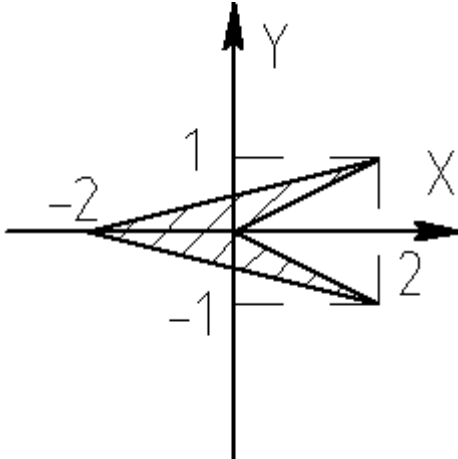


13. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких уводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній $Target$.

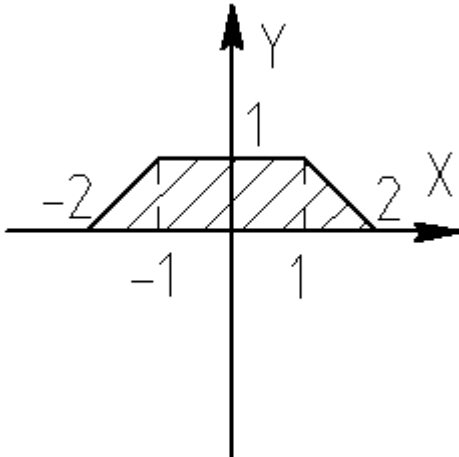
змінній Target.



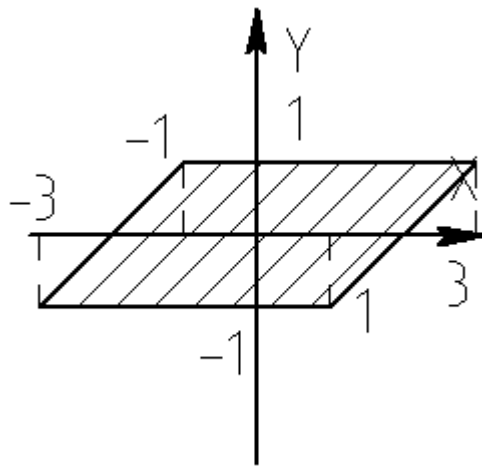
14. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній Target.



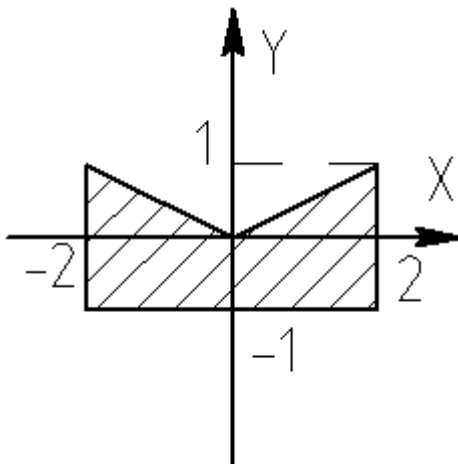
15. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній Target.



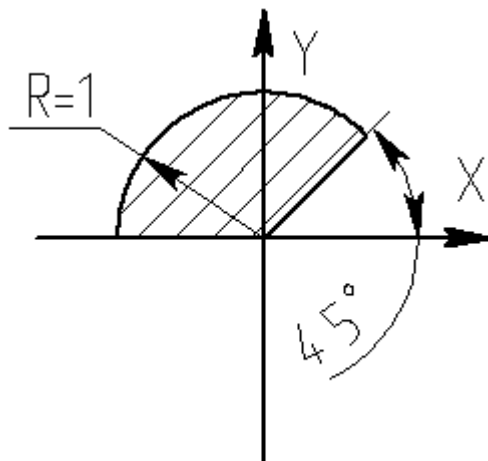
16. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній Target.



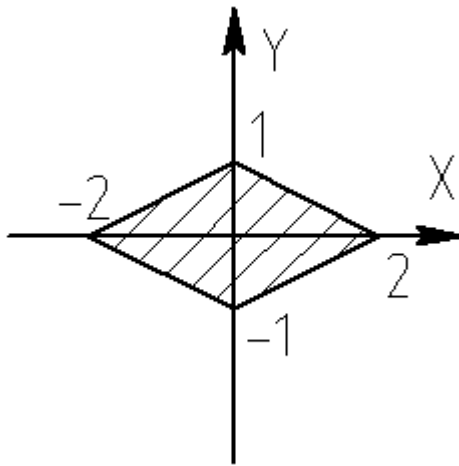
17. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній `Target`.



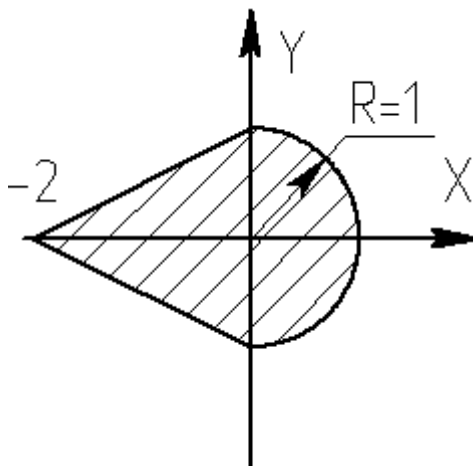
18. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній `Target`.



19. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній `Target`.



20. З'ясувати, чи потрапляє точка з координатами X, Y , значення яких вводяться з клавіатури до заштрихованої зони, зображеної на рисунку. Результат присвоїти логічній змінній `Target`.



Контрольні запитання

1. Наведіть приклади логічних змінних і констант. Який об'єм пам'яті вони займають?
2. Що таке логічний вираз? Поясніть на прикладі.
3. Які логічні операції ви знаєте? Наведіть приклади.
4. Які операції відношення Вам знайомі? Проілюструйте прикладом.
5. Які особливості використання дійсних і цілих чисел у логічних операціях?
6. Які особливості формування логічних виразів із використанням символічних і рядкових даних?
7. Прикладом проілюструйте пріоритет виконання логічних операцій.
8. Проілюструйте на прикладі відмінність логічних дій `not` та `and`.
9. Проілюструйте на прикладі відмінність логічних дій `or` та `xor`.
10. Який тип даних є результатом логічних операцій? Наведіть значення, які він може приймати.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 4

Розгалужений алгоритм. Оператор `IF`

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення розгалужених алгоритмів з використанням оператора перевірки умови IF.

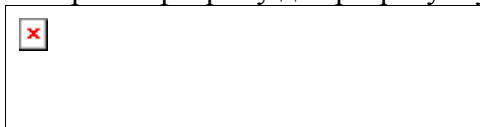
Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [2.1 з теми 12](#), уважно розглянути приклади [12.5](#) і [12.6](#). При вирішенні завдань цієї лабораторної роботи слід пам'ятати по таке:

- Інструкція IF використовується для вибору одного з двох напрямків подальшого ходу програми (послідовності інструкцій, які повинні виконуватися);
- Вибір послідовності інструкцій відбувається під час виконання програми в залежності від виконання або невиконання умови;
- Умова – це логічна змінна, константа або вираз, що може приймати значення true або false;
- Пам'ятайте, що if...then...else; – єдиний і неподільний оператор, тому не ставте крапку з комою перед else і не забувайте ставити цей службовий символ наприкінці;
- У разі застосування вкладених операторів перевірки умови else має відношення тільки до найближчого попереднього then;
- Якщо до складу логічного виразу входять дійсні числа, то не рекомендується використовувати операції чіткої рівності (=) або нерівності (<>). Замість цього краще аналізувати похибку, яку ви припускаєте для задачі. Наприклад, замість $X=0$, краще буде так: $X<0.1E-08$.

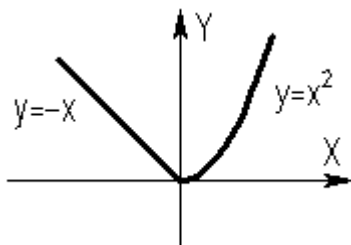
Варіанти завдань

1. З клавіатури вводяться дійсні числа A, B, C , які є коефіцієнтами квадратного рівняння виду $Ax^2+Bx+C=0$ (при $A<>0$). Вивести на екран одне з повідомлень: „Рівняння має два дійсних корені“, „Рівняння має один корінь“, „Рівняння не має дійсних коренів“.
2. Створити програму для розрахунку значення Y у відповідності до такої формули:

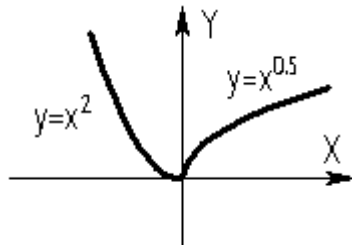


3. З клавіатури вводяться три числа. Якщо вони утворюють послідовність, що збільшується – вивести на екран повідомлення „Числова послідовність збільшується“. Якщо вони утворюють послідовність, що зменшується – вивести на екран повідомлення „Числова послідовність зменшується“. У протилежному випадку вивести на екран повідомлення „Невизначена числова послідовність“.
4. Перерозподілити значення дійсних змінних X і Y таким чином, щоб у X опинилося більше значення, а у Y – менше. Вивести на екран результат.
5. Змінній D присвоїти максимальне значення дійсних змінних A, B, C . Вивести на екран всі значення.
6. Змінній M присвоїти мінімальне значення змінних цілого типу I, J, K . Вивести на екран всі значення.
7. Змінній Z присвоїти максимальне значення дійсних змінних X, Y у випадку, коли $X<0$, і мінімальне значення X, Y у протилежному випадку. Вивести на екран всі значення.

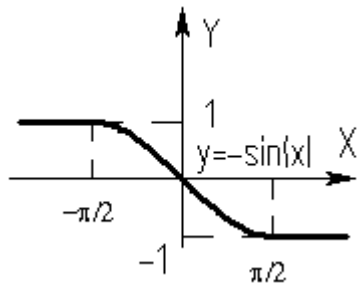
8. Змінній N присвоїти номер чверті координатної площини в яку потрапляє точка з координатами X, Y. Вивести на екран всі значення.
9. Значення змінних A, B, C, які вводяться з клавіатури поміняти місцями так, щоб стало: $A < B < C$. Вивести на екран початкову і кінцеву послідовність.
10. З клавіатури вводяться довільні дійсні числа L1, L2, L3. Проаналізувати, чи можна утворити трикутник з такими сторонами і вивести на екран одне із повідомлень: „Трикутник з такими сторонами не можна утворити”, „Це рівносторонній трикутник”, „Це не рівносторонній трикутник”.
11. З клавіатури вводяться довільні дійсні числа L1, L2, L3. Проаналізувати, чи можна утворити трикутник з такими сторонами і вивести на екран одне із повідомлень: „Трикутник з такими сторонами не можна утворити”, „Це рівнобічний трикутник”, „Це не рівнобічний трикутник”.
12. З клавіатури вводяться довільні дійсні числа L1, L2, L3. Проаналізувати, чи можна утворити трикутник з такими сторонами і вивести на екран одне із повідомлень: „Трикутник з такими сторонами не можна утворити”, „Це прямокутний трикутник”, „Це не прямокутний трикутник”.
13. Написати програму визначення вартості телефонної розмови з урахуванням 20% знижки у суботу і неділю.
14. Написати програму визначення загальної вартості покупки з урахуванням знижки. Знижка 3% надається коли вартість покупки перевищує 500 грн. і 5% – при вартості покупки понад 1000 грн.
15. Оптимальна вага людини визначається формулою: „Зріст (у см) - 100”. З клавіатури ввести вагу і зріст людини. Вивести на екран повідомлення: „Вага більша за норму”, „Вага менша за норму”, „Вага нормальна”. Вагу будемо вважати нормальною при відхиленні від точної норми $\pm 2,5$ кг.
16. Написати програму перевірки таблиці множення від 0 до 9. Пропонувати користувачеві вводити результат множення і виводити на екран повідомлення „Вірно” або „Помилка”.
17. Написати програму обчислення функції, що задана графічно.



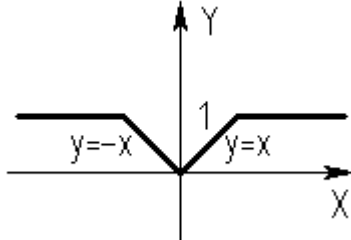
18. Написати програму обчислення функції, що задана графічно.



19. Написати програму обчислення функції, що задана графічно.



20. Написати програму обчислення функції, що задана графічно.



Контрольні запитання

1. Запишіть припустимі формати оператора перевірки умови `If`.
2. Якого типу може бути умова в операторі `If`?
3. Поясніть роботу спрощеного оператора `If`. Наведіть приклади.
4. Поясніть роботу повного оператора `If`. Наведіть приклади.
5. Як працюють вкладені оператори перевірки умови?
6. Чи можна перед `else` ставити крапку з комою? Чому?
7. Наведіть приклад використання оператора `If`, якщо треба, щоб після `then` виконувались декілька операторів?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 5

Розгалужений алгоритм. Оператор CASE

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення розгалужених алгоритмів з використанням оператора вибору `Case`.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [2.2 з теми 12](#), уважно розглянути приклад [12.7](#). При вирішенні завдань цієї лабораторної роботи слід пам'ятати по таке:

- Інструкція `Case` використовується для вибору одного з кількох подальших напрямків виконання програми;
- Вибір послідовності здійснюється під час виконання програми в залежності від

відповідності значення змінної-селектора одній з констант, яка передує певному оператору;

- Для визначення змінної-селектора можна використовувати тільки цілочислові (`integer`), символний (`char`) або логічний (`boolean`) типи;
- Константи можливих значень можуть записуватися як перелік значень (наприклад, `12, 'C'`) або як діапазон (наприклад, `0..9, 'A'..'Z'`);
- Списки можливих значень не повинні мати однакові значення. У випадку збіжності компілятор не буде видавати повідомлення про помилку, але виконає інструкцію, яку знайде першою;
- Якщо значення змінної-селектора не потрапило до списку можливих значень, тоді або жоден з операторів не буде виконуватися, або виконається оператор, що стоїть після `else` наприкінці оператора `case`;

Варіанти завдань

1. Скласти програму, яка буде реагувати на натискання на клавіатурі клавіш Y і N у будь-якому регістрі і, якщо була натиснута клавіша Y (y) – на екран виводити повідомлення „Так”, якщо була натиснута клавіша N (n) – виводити повідомлення „Ні”, якщо була натиснута будь-яка інша клавіша – виводити повідомлення „Вибір не зроблено”.
2. Скласти програму, яка буде реагувати на натискання на клавіатурі клавіш зі знаками математичних операцій ($+$, $-$, $*$, $/$). На екран повинні виводитись такі повідомлення: „Операція додавання”, „Операція віднімання”, „Операція множення”, „Операція ділення”. У разі натискання будь-якої іншої клавіші повинно виводитись на екран повідомлення „Уведений символ – не математична операція”.
3. Скласти програму, яка буде реагувати на натискання на клавіатурі символу і виводити на екран такі повідомлення: „Голосна латинська літера” – у випадку натискання будь-якої голосної латинської літери; „Приголосна латинська літера” – у випадку натискання приголосної латинської літери; „Уведений символ – не латинська літера” – у всіх інших випадках.
4. Скласти програму аналогічну до завдання 3, замінивши латинські літери українськими (російськими).
5. Використовуючи оператор `case`, визначити яке ціле число I уведено з клавіатури. Якщо $I < 0$ – на екран повинні виводитись такі повідомлення „Число менше нуля”, якщо $I > 0$ – „Число більше нуля”, якщо $I = 0$ – „Число дорівнює нулю”.
6. Скласти програму, яка буде реагувати на натискання на клавіатурі символу і виводити на екран такі повідомлення: „Натиснуто латинську літеру” – якщо клавіатура перебувала у стані вводу латинських символів; „Натиснуто літеру кирилиці” – якщо клавіатура перебувала у стані вводу символів кирилиці; „Невизначений символ” – в усіх інших випадках.
7. Скласти програму, яка буде відповідним чином реагувати на натискання символів „!”, „?”, „\$”, „%” і виводити на екран такі повідомлення: „Знак оклику”, „Знак питання”, „Символ долара”, „Символ відсотків”, „Невизначений символ” – в усіх інших випадках..
8. Скласти програму, яка буде дозволяти увести з клавіатури ціле число і визначити порядок числа. На екран повинні виводитись такі повідомлення: „Десятки тисяч”, „Тисячі”, „Сотні”, „Десятки”, „Одиниці”, „Нуль”.
9. Скласти програму, яка буде дозволяти по введеному номеру місяця визначати кількість днів у ньому. Вважати, що рік не високосний. Програма повинна відповідним чином реагувати на помилково введення номера місяця.
10. Скласти програму, яка буде дозволяти по введеному з клавіатури символу латиниці

визначити в якому реєстрі (верхньому чи нижньому) цей символ був уведений. На екран повинні видаватися такі повідомлення: „Символ – велика латинська літера“, „Символ – мала латинська літера“, „Невизначений символ“.

11. Скласти програму, яка буде дозволяти по введеному з клавіатури символу визначити чи є цей символ числом, літерою (латиниці та кирилиці), або іншим спеціальним символом. На екран повинні видаватися такі повідомлення: „Символ – число“, „Символ – літера“, „Тип символу не визначений“.
12. Скласти програму, яка після введення з клавіатури цілого числа в межах в 0 до 99, що позначає вартість у копійках, буде дописувати слово „копійка” у правильній формі (наприклад, 3 копійки, 7 копійок, 41 копійка).
13. Скласти програму, яка після введення з клавіатури цілого числа в межах в 0 до 100, що позначає вартість у гривнях, буде дописувати слово „гривня” у правильній формі (наприклад, 3 гривні, 7 гривень, 141 гривня).
14. Скласти програму, яка у відповідь на запитання про номер місяця буде виводити на екран інформацію про пору року або виводити повідомлення про те, що місяць з таким номером не існує.
15. Скласти програму, яка у відповідь на запитання про номер місяця буде виводити на екран такі повідомлення: „Перший учбовий семестр”, „Другий учбовий семестр”, „Літні канікули”, „Місяць з таким номером не існує“.
16. Скласти програму, яка у відповідь на запитання про екзаменаційну оцінку буде виводити на екран такі повідомлення: „Відмінно”, „Добре”, „Задовільно”, „Незадовільно”, „Неприпустима оцінка або символ“.
17. Скласти програму, яка у відповідь на запитання про номер дня тижня буде виводити такі повідомлення: „Робочий день”, „Вихідний день”, „Невірний номер дня тижня“.
18. Скласти програму, яка у відповідь на запитання про номер дня тижня буде виводити його текстову назву, або повідомлення про невірний номер дня тижня.
19. Скласти програму, яка у відповідь на запитання про номер місяця буде виводити його текстову назву, або повідомлення про невірний номер місяця.
20. Скласти програму, яка у відповідь на запитання про рік в межах від 2000 до 2010 буде виводити текстовий запис року, або повідомлення про те, що рік з таким номером не аналізується.

Контрольні запитання

1. Наведіть формат оператора Case.
2. Які типи даних можна використовувати в якості змінної-селектора?
3. В чому виявляється схожість і відмінність операторів Case і If?
4. Якими способами можна задавати можливі значення в операторі Case
5. Чи можуть списки можливих значень в операторі Case мати збіжні значення?
6. Чи можливий випадок, коли жоден з операторів у операторі Case не буде виконуватися?
7. Наведіть приклад використання оператора Case, якщо після двокрапки мають виконуватися декілька операторів.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 6

Циклічний алгоритм. Оператор FOR

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення циклічних алгоритмів з використанням оператора повторювання `For`.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [3.1 з теми 12](#), уважно розглянути приклади [12.8](#) і [12.9](#). При вирішенні завдань цієї лабораторної роботи слід пам'ятати по таке:

- Оператор `for` використовується для організації циклів з фіксованою кількістю повторювань;
- Параметр циклу – змінна одного з цілочислових, символьного або перелічуваного типу;
- Категорично забороняється змінювати значення параметру циклу всередині циклу;
- Кількість повторювань циклу визначається початковим і кінцевим значеннями параметру циклу і повинна бути визначеною до моменту виклику оператора;
- Початкове і кінцеве значення можуть задаватися як константи, змінні або вирази, але конкретне їхнє значення повинно бути визначено до моменту виклику оператора;
- Параметр циклу може збільшуватися або зменшуватися тільки на 1;
- Якщо всередину одного оператора циклу вкладений інший, то параметр циклу вкладеного оператора буде стільки разів змінюватися від початкового до кінцевого значення, скільки це вказано в оточуючому операторі циклу;
- Для встановлення кроку зміни, що відрізняється від 1 можна утворювати змінні, що будуть залежними від основного параметру циклу;

Варіанти завдань

Примітка. Для отримання відмінної оцінки таблиці повинні бути оформлені рамками і заголовками.

1. Вивести на екран таблицю $\cos(x) - \sin(y)$ при зміні y від 0° до 90° з кроком 15° (виведення по горизонталі) і зміні x від 0° до 180° з кроком 30° (виведення по вертикалі).
2. Вивести на екран таблицю $1 + \sin(x)$, $1 - \sin(x)$, $1 + \operatorname{tg}(x)$, $1 - \operatorname{tg}(x)$ (виведення по горизонталі) при зміні аргументу від 0° до 180° з кроком 15° (виведення по вертикалі). В програмі передбачити неможливість обчислення $\operatorname{tg}(90^\circ)$.
3. Вивести на екран таблицю $\sin(x) + \sin(y)$ при зміні y від 0° до 90° з кроком 30° (виведення по горизонталі) і зміні x від 0° до 90° з кроком 10° (виведення по вертикалі).
4. Вивести на екран таблицю $\ln(x)$, e^x при зміні аргументу від 0 до e з кроком $e/20$. В програмі передбачити неможливість обчислення $\ln(0)$.
5. Вивести на екран таблицю $\operatorname{tg}(x)$, $\operatorname{ctg}(x)$ (по горизонталі) при зміні аргументу від 0° до 90° з кроком 5° (виведення по вертикалі). В програмі передбачити неможливість ділення на 0.
6. Вивести на екран таблицю $1/\sin(x)$, $1/\cos(x)$, $1/(\sin(x) \cdot \cos(x))$ (по горизонталі), при зміні аргументу від 0° до 180° з кроком 10° (виведення по вертикалі). В програмі передбачити неможливість ділення на 0.
7. Вивести на екран таблицю множення у вигляді квадрату Піфагора. Таблицю вивести при зменшенні множників від 10 до 1.
8. Вивести на екран таблицю множення на 7 при зростанні першого множника від 1 до 10.
9. Вивести на екран таблицю множення на 8 при зменшенні першого множника від 10 до 1.
10. Вивести на екран таблицю квадратів перших N цілих додатних непарних чисел.

Кількість таких чисел N повинна вводитись під час виконання програми.

11. Вивести на екран таблицю степенів двійки від 0 до 10.
12. Вивести на екран таблицю значень функції $y = -2,4x^2 + 5x - 3$ у діапазоні від -2 до 2 з кроком 0,5.
13. Вивести на екран середнє арифметичне значення довільної послідовності N дійсних чисел. Кількість чисел і їхні значення вводяться з клавіатури.
14. Вивести таблицю вартості яблук в діапазоні від 50г до 1кг з кроком 50г. Вартість 1кг яблук повинна вводитися під час роботи програми.
15. Вивести на екран таблицю значень функції $y = |x-2| + |x-1|$ у діапазоні від -4 до 4 з кроком 0,5.
16. Вивести на екран таблицю $\cos(x) + \sin^2(y)$ при зміні y від 0° до 90° з кроком 30° (виведення по горизонталі) і зміні x від 0° до 180° з кроком 15° (виведення по вертикалі).
17. Вивести на екран таблицю $\sin(x)$, $\cos(x)$, $\operatorname{tg}(x)$, $\operatorname{ctg}(x)$ (виведення по горизонталі) при зміні аргументу від 0° до 180° з кроком 10° (виведення по вертикалі). В програмі передбачити неможливість обчислення $\operatorname{tg}(90^\circ)$, $\operatorname{ctg}(0^\circ)$, $\operatorname{ctg}(180^\circ)$.
18. Вивести на екран таблицю $\sin(x) \cdot \sin(y)$ при зміні y від 0° до 90° з кроком 15° (виведення по горизонталі) і зміні x від 0° до 90° з кроком 15° (виведення по вертикалі).
19. Вивести на екран таблицю x^a , a^x при зміні a від 0 до e з кроком $e/20$. Значення дійсної змінної x повинно вводитися з клавіатури.
20. Вивести на екран таблицю x^n при зміні n від 1 до 5 (виведення по горизонталі) та при зміні x від 1 до 9 з кроком 2 (виведення по вертикалі).

Контрольні запитання

1. Вкажіть можливі формати оператора циклу `for`.
2. Якого типу може бути параметр циклу?
3. Наведіть приклад застосування в циклах параметру символьного типу.
4. Які підрахувати кількість повторів оператора циклу `for`?
5. Якими способами можна визначати початкове і кінцеве значення параметру циклу?
6. Як слід поступити, якщо зміна в межах циклу повинна відбуватися з кроком, що відрізняється від 1 або -1?
7. Як працюють вкладені один в один оператори циклу?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 7

Циклічний алгоритм. Оператори Repeat...Until та While

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення циклічних алгоритмів з використанням операторів повторювання Repeat...Until та While.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [3.2](#) і [3.3](#) з [теми 12](#), уважно розглянути приклади [12.10](#), [12.11](#) і [12.12](#). При вирішенні завдань цієї лабораторної роботи слід пам'ятати

по такє:

- Оператори `repeat...until` і `while` використовуються для організації циклів з наперед невідомою кількістю повторів;
- Оператор `repeat...until` виконується до тих пір, поки умова, що слідує за `until` залишається невірною;
- Оператор `while` виконується до тих пір поки умова, що слідує за `while` залишається вірною;
- Оператор `repeat...until` виконується хоча б один раз, оператор `while` може не виконуватися жодного разу;
- Якщо тіло оператора циклу `while` складається більше ніж з одного оператора, слід застосовувати складений оператор `begin...end`;
- Всередині операторів циклу `repeat...until` і `while` обов'язково повинна бути інструкція, що впливає на умову завершення циклу. В іншому випадку ви ризикуєте отримати нескінченний цикл;
- Оператори циклу `repeat...until` і `while`, як правило, використовуються для організації наближених розрахунків, задач пошуку і обробки даних, які вводяться з клавіатури або з файла.

Варіанти завдань

1. Напишіть програму, основою якої є цикл, який виводить на екран запит і завершується тільки тоді, коли введене числове значення попадає у діапазон визначений константами `Min`, `Max`. Перед кожним повторюванням очищувати екран.
2. Напишіть програму, основою якої є цикл, який виводить на екран запит і завершується тільки тоді, коли користувач увів один із символів `Y`, `y`, `N`, `n`.
3. Напишіть програму, основою якої є цикл, який виводить на екран запитання (придумайте самостійно) і завершується тільки у випадку вірної відповіді на запитання. У разі невірної відповіді слід очищувати екран, виводити повідомлення про помилку і повторювати запитання.
4. Напишіть програму, яка з точністю, що введена з клавіатури розраховує число π . Для цього скористайтеся тим, що значення часткової суми послідовності $1 - 1/3 + 1/5 - 1/7 + 1/9 + \dots$ при підсумовуванні достатньо великої кількості членів наближується до $\pi/4$.

5. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \ln(x)$, ($x > 0$). Для цього скористайтеся такою формулою:

$$2 \sum_{n=0}^{\infty} \frac{(x-1)^{2n+1}}{(2n+1)(x+1)^{2n+1}} = 2 \left(\frac{x-1}{x+1} + \frac{(x-1)^3}{3(x+1)^3} + \frac{(x-1)^5}{5(x+1)^5} + \dots \right)$$

6. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \ln(x)$, ($0 < x \leq 2$). Для цього скористайтеся такою формулою:

$$\sum_{n=1}^{\infty} (-1)^{n+1} \frac{(x-1)^n}{n} = (x-1) - \frac{(x-1)^2}{2} + \frac{(x-1)^3}{3} - \frac{(x-1)^4}{4} + \dots$$

7. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \ln(x)$, ($x > 1/2$). Для цього скористайтеся такою формулою:

$$\sum_{n=1}^{\infty} \frac{(x-1)^n}{nx^n} = \frac{(x-1)}{x} + \frac{(x-1)^2}{2x^2} - \frac{(x-1)^3}{3x^3} + \dots$$

8. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення

функції $y = \ln(1+x)$, ($|x| < 1$). Для цього скористайтеся такою формулою:

$$\sum_{n=1}^{\infty} (-1)^{n+1} \frac{x^n}{n} = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots$$

9. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \ln(1-x)$, ($|x| < 1$). Для цього скористайтеся такою формулою:

$$-\sum_{n=1}^{\infty} \frac{x^n}{n} = -\left(x + \frac{x^2}{2} + \frac{x^3}{3} + \frac{x^4}{4} + \dots\right)$$

10. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \ln\left(\frac{1+x}{1-x}\right)$, ($|x| < 1$). Для цього скористайтеся такою формулою:

$$2 \sum_{n=0}^{\infty} \frac{x^{2n+1}}{(2n+1)} = 2 \left(x + \frac{x^3}{3} + \frac{x^5}{5} + \frac{x^7}{7} + \dots \right)$$

11. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \ln\left(\frac{x+1}{x-1}\right)$, ($|x| > 1$). Для цього скористайтеся такою формулою:

$$2 \sum_{n=0}^{\infty} \frac{1}{(2n+1)x^{2n+1}} = 2 \left(\frac{1}{x} + \frac{1}{3x^3} + \frac{1}{5x^5} + \frac{1}{7x^7} + \dots \right)$$

12. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \operatorname{arctg}(x)$, ($|x| < 1$). Для цього скористайтеся такою формулою:

$$\sum_{n=0}^{\infty} (-1)^n \frac{x^{2n+1}}{2n+1} = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$$

13. Напишіть програму, яка з точністю, що введена з клавіатури розраховує значення функції $y = \operatorname{arctg}(x)$, ($|x| < 1$). Для цього скористайтеся такою формулою:

$$\frac{\pi}{2} + \sum_{n=0}^{\infty} (-1)^{n+1} \frac{x^{2n+1}}{2n+1} = \frac{\pi}{2} - \left(x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots \right)$$

14. Напишіть програму, яка з точністю, що вводиться з клавіатури знаходить суму ряду $1 + 1/2 + 1/3 + 1/4 + \dots$

15. Напишіть програму, яка з точністю 0,1% знаходить суму ряду $1 + 1/2 + 1/3 + 1/4 + \dots$

16. Напишіть програму, яка з точністю, що вводиться з клавіатури знаходить суму ряду $1 + 1/2^2 + 1/3^2 + 1/4^2 + \dots$

17. Напишіть програму, яка з точністю 0,05% знаходить суму ряду $1 + 1/2^2 + 1/3^2 + 1/4^2 + \dots$

18. Напишіть програму, яка з точністю, що вводиться з клавіатури знаходить суму ряду $(1 - 1/2^2) \cdot (1 - 1/3^2) \cdot (1 - 1/4^2) + \dots$

19. Напишіть програму, яка з точністю 0,01% знаходить суму ряду $(1 - 1/2^2) \cdot (1 - 1/3^2) \cdot (1 - 1/4^2) + \dots$

20. Напишіть програму для розрахунку значення функції $y = \cos(x) + \cos(x^2) + \cos(x^3) + \dots + \cos(x^n)$, коли значення кута x (у радіанах) і кількість членів суми n вводяться з клавіатури.

Контрольні запитання

1. Вкажіть формат оператора циклу `repeat...until`.
2. Вкажіть формат оператора циклу `while`.
3. До яких пір буде виконуватися оператор циклу `repeat...until`?
4. До яких пір буде виконуватися оператор циклу `while`?
5. Що є спільним і у чому полягає різниця між операторами циклу `repeat...until` і `while`?
6. Скільки операторів може знаходитися всередині оператора `repeat...until`?
7. Скільки операторів може знаходитися у тілі оператора циклу `while`?
8. Яким чином можна впливати на умову завершення операторів циклу `repeat...until` і `while`? Для чого це слід робити?
9. Для вирішення яких задач переважно більшістю використовуються оператори циклу `repeat...until` і `while`?
10. Як працюють вкладені один в один оператори циклу?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 8

Структуризація програм. Функції

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок структуризації програм з використанням функцій користувача.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [1 і 3 з теми 13](#), уважно розглянути приклади [13.1](#), [13.2](#) і [13.3](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Перед використанням у блоку операторів програми нова функція користувача обов'язково повинна бути описаною у розділі описування;
- Серед інструкцій, що виконуються у тілі функції, обов'язково повинна бути інструкція присвоєння функції певного значення. Як правило така інструкція знаходиться наприкінці блоку операторів функції;
- Параметри функції можуть бути параметрами-значеннями і параметрами-змінними. В останньому випадку перед ними ставиться слово `var`;
- Функцію можна використовувати безпосередньо у виразах основної програми;
- При викликанні функції користувача в основній програмі параметри, що передаються у функцію повинні мати такий саме тип, як при описуванні функції;
- Рекомендується для передавання даних у функцію використовувати тільки параметри і не посилатися всередині функції на глобальні змінні і константи;
- Пам'ятайте про те, що у випадку збіжності імен локальної і глобальної змінної або константи, глобальний елемент не буде доступним всередині функції, він буде перекритим локальним.

Варіанти завдань

1. Написати функцію, яка знаходить гіперболічний синус і реалізується формулою:

$$y = \operatorname{sh} x = \frac{e^x - e^{-x}}{2}$$

2. Написати функцію, яка знаходить гіперболічний косинус і реалізується формулою:

$$y = \operatorname{ch} x = \frac{e^x + e^{-x}}{2}$$

3. Написати функцію, яка знаходить гіперболічний тангенс і реалізується формулою:

$$y = \operatorname{th} x = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

4. Написати функцію, яка знаходить гіперболічний котангенс і реалізується формулою:

$$y = \operatorname{cth} x = \frac{e^x + e^{-x}}{e^x - e^{-x}}, \quad x \neq 0$$

5. Написати функцію, яка знаходить гіперболічний секанс і реалізується формулою:

$$y = \operatorname{sch} x = \frac{2}{e^x + e^{-x}}$$

6. Написати функцію, яка знаходить гіперболічний синус і реалізується формулою:

$$y = \operatorname{csch} x = \frac{e^x - e^{-x}}{e^x + e^{-x}}, \quad x \neq 0$$

7. Написати функцію, яка знаходить обернений гіперболічний синус і реалізується формулою:

$$y = \operatorname{arsh} x = \ln(x + \sqrt{x^2 + 1})$$

8. Написати функцію, яка знаходить обернений гіперболічний косинус і реалізується формулою:

$$y = \operatorname{arch} x = \ln(x - \sqrt{x^2 - 1}), \quad (\text{для } x \geq 1 \text{ и } -\infty < y \leq 0)$$

9. Написати функцію, яка знаходить обернений гіперболічний тангенс і реалізується формулою:

$$y = \operatorname{arth} x = \ln \sqrt{\frac{1+x}{1-x}} = \frac{1}{2} \ln \frac{1+x}{1-x}, \quad (\text{при } |x| < 1)$$

10. Написати функцію, яка знаходить обернений гіперболічний тангенс і реалізується формулою:

$$y = \operatorname{arth} x = \ln \sqrt{\frac{x+1}{x-1}} = \frac{1}{2} \ln \frac{x+1}{x-1}, \quad (\text{при } |x| > 1)$$

11. Написати функцію розрахунку довжини медіани, яка проведена до сторони a і реалізується формулою:

$$m_a = \frac{\sqrt{2(b^2 + c^2) - a^2}}{2}$$

12. Написати функцію розрахунку довжини бісектриси кута a , яка реалізується формулою:

$$l_a = \frac{\sqrt{bc[(b+c)^2 - a^2]}}{b+c}$$

13. Написати функцію розрахунку площі трикутника, реалізувавши її за формулою Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)}, \quad \text{де } p = (a+b+c)/2$$

14. Написати функцію розрахунку площі вписаного у коло прямокутника за формулою:

$$S = \sqrt{(p-a)(p-b)(p-c)(p-d)}, \quad \text{де } p = (a+b+c+d)/2$$

15. Написати функцію розрахунку об'єму прямого зрізаного конусу висотою h , радіусами більшої R і меншої r основи за формулою:

$$V = \pi h (R^2 + r^2 + Rr) / 3$$

16. Написати функцію розрахунку радіусу описаного навколо трикутника з кутами α, β, γ і на півпериметром p кола за формулою:

$$R = \frac{P}{4 \cos \frac{\alpha}{2} \cos \frac{\beta}{2} \cos \frac{\gamma}{2}}$$

17. Написати функцію розрахунку вписаного у трикутник зі сторонами a, b, c кола за формулою:

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

18. Реалізувати логічну функцію, яка вказує існує чи ні трикутник із довжинами сторін a, b, c .
19. Реалізувати логічну функцію, яка вказує існує чи ні трикутник, координати вершин якого $x_1, y_1, x_2, y_2, x_3, y_3$.
20. Реалізувати логічну функцію, яка вказує, чи потрапило дійсне число X у діапазон $\text{Min} \dots \text{Max}$.

Контрольні запитання

1. Вкажіть, з якою метою у Pascal реалізовано механізм функцій користувача?
2. Чим функція відрізняється від процедури?
3. У чому полягає відмінність між стандартними функціями і функціями користувача?
4. Які стандартні математичні функції Pascal ви знаєте?
5. Чи можна у програмі використовувати змінні і константи, які були описані у функції користувача?
6. Чи можна використовувати у функції змінні і константи, які були описані у програмі?
7. Що станеться, коли в програмі і функції описати змінні з однаковими іменами?
8. Чи можна всередині функції викликати її саму?
9. Чи можуть тип параметрів і тип функції бути однаковими? Різними? Наведіть приклади.
10. В яких випадках замість параметра функції можна підставляти безпосередньо деяке значення, а коли лише ідентифікатор змінної?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 9

Структуризація програм. Процедури

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок структуризації програм з використанням процедур користувача.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити питання [1 і 3 з теми 13](#), уважно розглянути приклади [13.4](#), [13.5](#) і [13.6](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

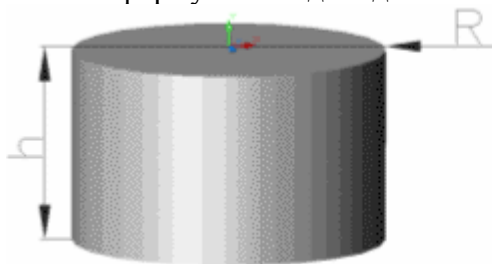
- Перед використанням у блоку операторів програми нова процедура користувача обов'язково повинна бути описаною у розділі описування;
- Параметри процедури можуть бути параметрами-значеннями і параметрами-змінними. В останньому випадку перед ними ставиться слово `var`;
- Виклик процедури здійснюється окремим оператором програми шляхом вказування

імені процедури і фактичних параметрів, що передаються;

- При викликанні процедури користувача в основній програмі параметри, що передаються у процедуру повинні мати такий саме тип, як при описуванні;
- Рекомендується для передавання даних у процедуру використовувати тільки параметри і не посилатися всередині процедури на глобальні змінні і константи;
- Пам'ятайте про те, що у випадку збіжності імен локальної і глобальної змінної або константи, глобальний елемент не буде доступним всередині процедури, він буде перекритим локальним.

Варіанти завдань

1. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V кругового прямого циліндра. Схема і формули наведені далі.

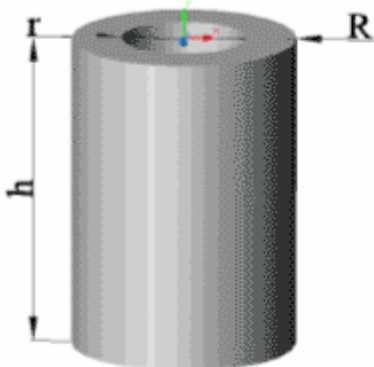


$$M = 2 \cdot \pi \cdot R \cdot h$$

$$S = 2 \cdot \pi \cdot R \cdot (R + h)$$

$$V = \pi \cdot R^2 \cdot h$$

2. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V циліндричної труби. Схема і формули наведені далі.

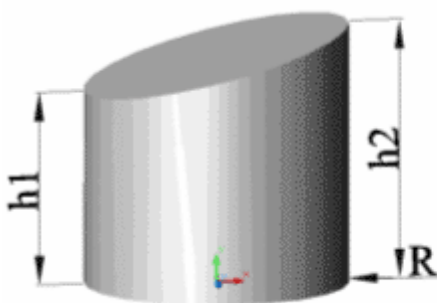


$$M = 2 \cdot \pi \cdot h \cdot (R + r)$$

$$S = 2 \cdot \pi \cdot R \cdot (R + r + h)$$

$$V = \pi \cdot h \cdot (R^2 - r^2)$$

3. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V відсіченого циліндра. Схема і формули наведені далі.

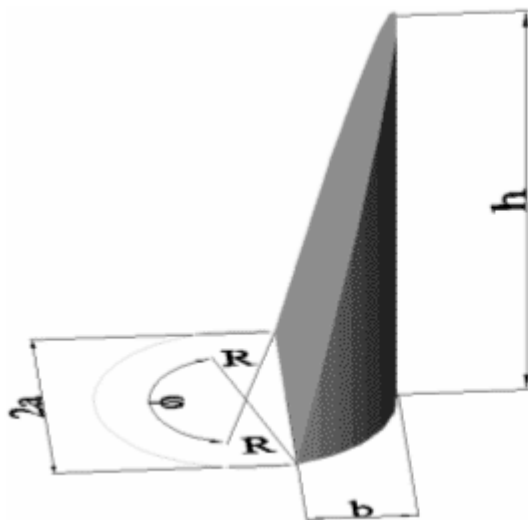


$$M = \pi \cdot R \cdot (h_1 + h_2)$$

$$S = \pi \cdot R \cdot \left[h_1 + h_2 + R + \sqrt{R^2 + \left(\frac{h_1}{2}\right)^2} \right]$$

$$V = \pi \cdot R^2 \cdot \left(\frac{h_1 + h_2}{2} \right)$$

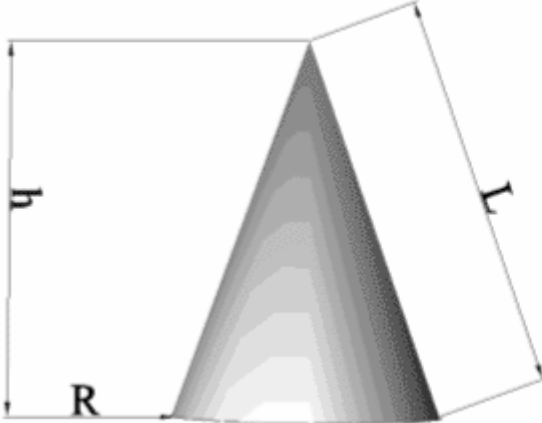
4. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V відрізка циліндра („копито”). Схема і формули наведені далі.



$$M = 2 \cdot R \cdot h \cdot \frac{[(b - R) \cdot \varphi + a]}{b}$$

$$V = h \cdot \frac{[a \cdot (3 \cdot R^2 - a^2) + 3 \cdot R^2 \cdot (b - a)]}{3 \cdot b}$$

5. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V кругового прямого конусу. Схема і формули наведені далі.

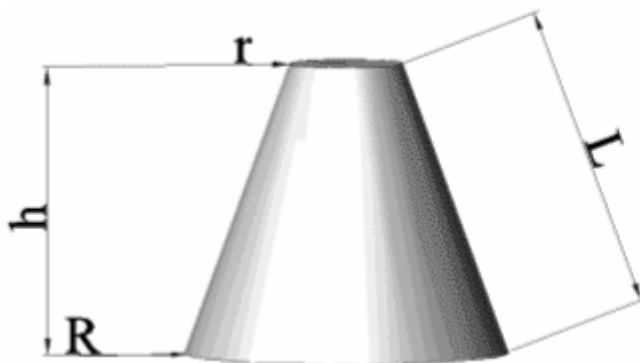


$$M = \pi \cdot R \cdot \sqrt{R^2 + h^2}$$

$$S = \pi \cdot R \cdot (R + \sqrt{R^2 + h^2})$$

$$V = \frac{\pi \cdot R^2 \cdot h}{3}$$

6. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V відсіченого прямого конусу. Схема і формули наведені далі.



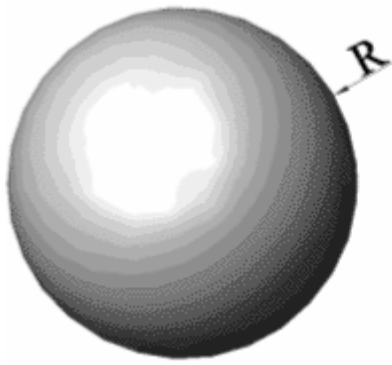
$$l = \sqrt{h^2 + (R - r)^2}$$

$$M = \pi \cdot l \cdot (R + r)$$

$$S = \pi \cdot (R + l) \cdot (R + r)$$

$$V = \frac{\pi \cdot h \cdot (R^2 + r^2 + R \cdot r)}{3}$$

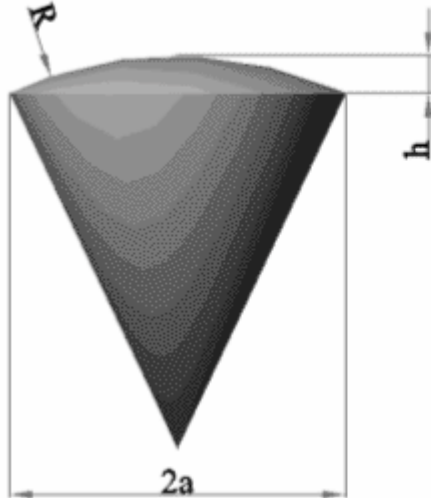
7. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі поверхні S , об'єму V та маси m сфери. Схема і формули наведені далі.



$$S = 4 \cdot \pi \cdot R^2$$

$$V = \frac{4 \cdot \pi \cdot R^3}{3}$$

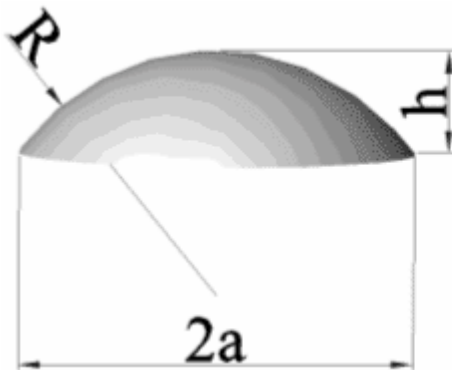
8. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі поверхні S , об'єму V та маси m сферичного сектора. Схема і формули наведені далі.



$$S = \pi \cdot R \cdot (2 \cdot h + a)$$

$$V = \frac{2 \cdot \pi \cdot R^2 \cdot h}{3}$$

9. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V сферичного сегмента. Схема і формули наведені далі.



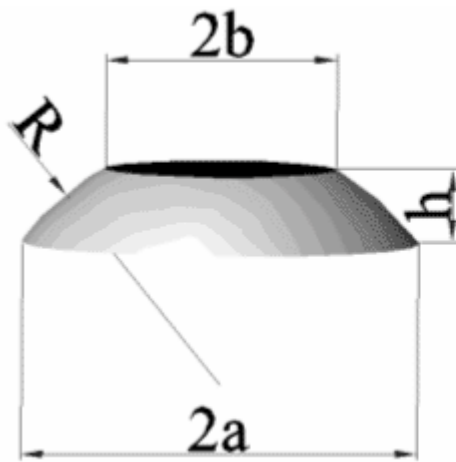
$$a = h \cdot (2 \cdot R - h)$$

$$M = 2 \cdot \pi \cdot R \cdot h$$

$$S = \pi \cdot (h^2 + 2 \cdot a^2)$$

$$V = \frac{\pi \cdot h^2 \cdot (3 \cdot R - h)}{3}$$

10. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі бічної поверхні M , повної площі S та об'єму V шару сфери. Схема і формули наведені далі.



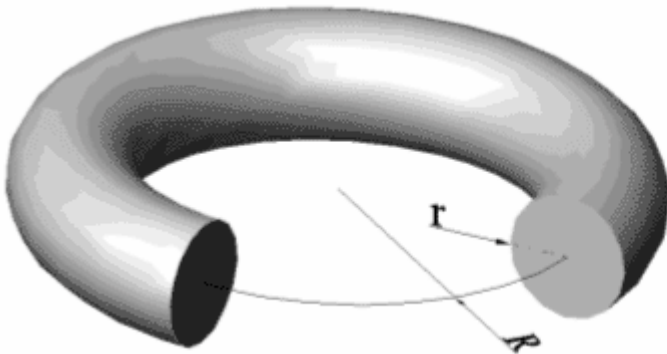
$$R^2 = a^2 + \left[\frac{a^2 - b^2 - h^2}{2 \cdot h} \right]^2$$

$$M = 2 \cdot \pi \cdot R \cdot h$$

$$S = \pi \cdot (2 \cdot R \cdot h + a^2 + b^2)$$

$$V = \frac{\pi \cdot h \cdot (3 \cdot a^2 + 3 \cdot b^2 + h^2)}{6}$$

11. Написати програму в якій описати і використати процедуру обчислення і виведення на екран площі поверхні S , об'єму V та маси m тора. Схема і формули наведені далі.



$$S = 4 \cdot \pi^2 \cdot R \cdot r$$

$$V = 2 \cdot \pi^2 \cdot R \cdot r^2$$

12. Написати програму в якій описати і використати процедуру `ReadStringXY` зчитування з клавіатури рядка тексту. Через параметри в процедуру передавати `Col` – колонку і `Row` – рядок у позиції яких повинен починатися запит `Prompt`. Уведений результат зберігати у змінній `St`.
13. Написати програму в якій описати і використати процедуру `ReadIntegerXY` зчитування з клавіатури цілого числа. Через параметри в процедуру передавати `Col` – колонку і `Row` – рядок у позиції яких повинен починатися запит `Prompt`. Уведений результат зберігати у змінній `I`.
14. Написати програму в якій описати і використати процедуру `ReadRealXY` зчитування з клавіатури дійсного числа. Через параметри в процедуру передавати `Col` – колонку і `Row` – рядок у позиції яких повинен починатися запит `Prompt`. Уведений результат зберігати у змінній `R`.
15. Написати програму в якій описати і використати процедуру `WriteStringXY` виведення на екран рядка тексту. Через параметри в процедуру передавати `Col` – колонку і `Row` – рядок у позиції яких повинно починатися виведення тексту `St`, а також `Leng` – загальна довжина виведення тексту.
16. Написати програму в якій описати і використати процедуру `WriteIntegerXY` виведення на екран цілого числа. Через параметри в процедуру передавати `Col` – колонку і `Row` – рядок у позиції яких повинно починатися виведення числа `I`, а також `Leng` – загальна довжина виведення числа.
17. Написати програму в якій описати і використати процедуру `WriteRealXY` виведення на екран цілого числа. Через параметри в процедуру передавати `Col` – колонку і `Row` – рядок у позиції яких повинно починатися виведення числа `R`, `Leng` – загальна довжина виведення числа, `Dec` – кількість символів дробової частини.
18. Написати програму в якій описати і використати процедуру `MyWindow` виведення на екран простої рамки (вікна), розміри якої визначаються `X1`, `Y1` – координатами верхнього лівого кута і `X2`, `Y2` – координатами правого нижнього кута.

19. Написати програму в якій описати і використати процедуру `WindowHor` виведення на екран рамки (вікна), розділеної горизонтальною лінією приблизно посередині. Розміри рамки визначаються $X1, Y1$ – координатами верхнього лівого кута і L, H – довжиною і висотою рамки.
20. Написати програму в якій описати і використати процедуру `WindowVer` виведення на екран рамки (вікна), розділеної вертикальною лінією приблизно посередині. Розміри рамки визначаються $X1, Y1$ – координатами верхнього лівого кута і L, H – довжиною і висотою рамки.

Контрольні запитання

1. Вкажіть, з якою метою у Pascal реалізовано механізм процедур користувача?
2. Чим процедура відрізняється від функції?
3. У чому полягає відмінність між стандартними процедурами і процедурами користувача?
4. Які стандартні процедур Pascal ви знаєте?
5. Чи можна у програмі використовувати змінні і константи, які були описані у процедурі користувача?
6. Чи можна використовувати у процедурі змінні і константи, які були описані у програмі?
7. Що станеться, коли в програмі і функції описати змінні з однаковими іменами?
8. Чи можна посилатися у процедурі на іншу процедуру, яка буде визначатися далі?
9. Чим відрізняються параметри-значення від параметрів-змінних? Наведіть приклади?
10. Чи можуть формальні та фактичні параметри мати однаковий тип даних? Різний?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 10

Масиви

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення і використання масивів

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 14](#), уважно розглянути приклади [14.1](#), [14.2](#), [14.3](#), [14.4](#), [14.5](#) і [14.6](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Масив – структурований тип даних, який є сукупністю елементів одного типу;
- Масив необхідно описати в розділі описування, обов'язково вказавши ім'я масиву, діапазон індексів масиву і тип його елементів;
- Для встановлення діапазону індексів масиву зручно використовувати іменовані константи;
- Дістатися до конкретного елемента масиву можна вказавши ім'я масиву і його індексу (індексів). Індекс (індекси) елементів масиву вказуються у квадратних дужках;
- Значення індексу можна передавати як константу, змінну або вираз, ти яких повинен відповідати типові індексу;
- Для масивів присвоювати значення, виводити на екран та зчитувати з клавіатури можна лише поелементно.

Варіанти завдань

1. Таблиця футбольного чемпіонату задається квадратною матрицею розмірністю 16×16 , в якій всі елементи на головній діагоналі дорівнюють нулю, а залишок заповнений за такими правилами: 3 – перемога, 1 – нічия, 0 – поразка. Визначити команди, що зайняли 1, 2 і 3 місця. Вивести на екран їхні номери і кількість набраних очок. Початковий стан матриці сформувати за допомогою операторів присвоювання. Орієнтовний стан матриці наведений далі.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	0	0	3	3	0	1	1	0	0	0	3	1	1	3	0	3
2	3	0	0	0	0	1	1	0	1	0	3	0	0	1	1	1
3	0	3	0	1	1	3	0	1	1	1	1	3	3	1	0	0
4	1	1	1	0	1	1	1	1	3	3	3	0	0	0	0	1
5	1	3	1	3	0	1	3	1	3	1	3	1	3	1	3	1
6	3	0	3	0	3	0	3	0	3	0	3	0	3	0	3	0
7	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	1	1	1	1	1	1	1	1	0	3	3	3	3	3	3	3
10	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
11	0	0	3	3	3	1	1	1	3	3	0	3	3	1	1	3
12	3	3	3	3	3	3	3	3	3	3	3	0	3	3	3	3
13	1	1	3	3	3	1	1	1	0	0	0	0	0	1	1	1
14	0	0	0	0	1	1	1	1	1	1	3	3	3	0	0	0
15	3	3	1	1	3	3	3	3	1	1	3	3	3	3	0	3
16	0	1	3	0	1	3	0	1	3	0	1	3	0	1	3	0

2. За умовою завдання 1 визначити команди, які пройшли чемпіонат без поразок. Вивести на екран номери команд і кількість набраних ними очок. Початковий стан матриці визначити через типізовану константу-матрицю.
3. За умовою завдання 1 визначити номери команд, які мають більше перемог, ніж поразок. Вивести на екран номери цих команд, кількість перемог і кількість поразок для кожної з них. Початковий стан матриці сформувати за допомогою операторів присвоювання.
4. За умовою завдання 1 визначити дві команди, які набрали найменше очок і вибувають з наступного чемпіонату. Вивести на екран номери команд і кількість набраних ними очок. Початковий стан матриці визначити через типізовану константу-матрицю.
5. За умовою завдання 1 визначити, чи існують команди, які виграли більше половини ігор. Якщо „так” – вивести на екран номери цих команд і кількість вигравів. Початковий стан матриці сформувати за допомогою операторів присвоювання.
6. За умовою завдання 1 розташувати команди у порядку зменшенні кількості очок. Вивести на екран місце, яке зайняла команда, її номер та кількість набраних очок. Початковий стан матриці визначити через типізовану константу-матрицю.
7. За умовою завдання 1 визначити, чи є такі команди, які не виграли жодної ігри. Вивести на екран номери таких команд і кількість набраних ними очок. Початковий стан матриці сформувати за допомогою операторів присвоювання.
8. За умовою завдання 1 визначити, чи є такі команди, які не програли жодної ігри. Вивести на екран номери таких команд і кількість набраних ними очок. Початковий стан матриці визначити через типізовану константу-матрицю.
9. За умовою завдання 1 визначити, чи є такі команди, які грали виключно у нічию. Вивести на екран номери таких команд. Початковий стан матриці сформувати за допомогою операторів присвоювання.
10. Є цілочислова матриця розмірністю 8×8 , яка характеризує взаємні обмеження рухливості між деталями у складальній одиниці. Елементами матриці можуть бути 0 (означає відсутність обмеження) і 1 (означає наявність обмеження). Причому рядки

вказують на обмеження у позитивному напрямку, а стовпчики – у від'ємному. Визначити, яка деталь або деталі мають найбільшу кількість обмежень з боку інших деталей. Вивести на екран номери таких деталей і номери деталей, що їх обмежують. Початковий стан матриці сформувати за допомогою операторів присвоювання. Орієнтовний вид матриці наведений далі.

X	1	2	3	4	5	6	7	8
1	0	1	0	1	0	0	1	0
2	1	0	0	0	0	1	0	1
3	0	1	0	0	0	0	1	0
4	0	0	0	0	0	0	0	0
5	0	1	0	0	0	0	0	0
6	1	1	0	0	0	0	1	1
7	1	1	1	1	1	1	0	1
8	0	1	0	1	0	1	1	0

- За умовою завдання 10 визначити, чи існують деталі які не мають обмежень у позитивному або від'ємному напрямках. Якщо такі деталі є, тоді вивести на екран номери цих деталей і напрямок (+ або -) відсутності обмеження. Початковий стан матриці визначити через типізовану константу-матрицю.
- За умовою завдання 10 визначити чи існують такі деталі, які обмежені усіма іншими деталями складальної одиниці. Якщо такі деталі є, тоді вивести на екран номери цих деталей і напрямок (+ або -) повного обмеження. Початковий стан матриці сформувати за допомогою операторів присвоювання.
- Є цілочислова квадратна матриця розміром 15×15 . Знайти середнє арифметичне всіх рядків і всіх стовпчиків матриці. Вивести на екран отримані значення. Початковий стан матриці сформувати з допомогою операторів присвоювання.
- Є дійсна матриця розміром 8×10 . Замінити всі її ненульові елементи таким чином: якщо елемент додатний – „1”, якщо елемент від'ємний – „-1”. Вивести на екран початкову і нову матриці, а також кількість від'ємних, додатних і нульових елементів. Початковий стан матриці сформувати з допомогою типізованої константи-матриці.
- Є цілочислова матриця розміром 6×12 . Знайти кількість і місце знаходження таких елементів, які більші за елемент, що знаходиться ліворуч у рядку, і менші за елемент, що знаходиться праворуч у рядку. Вивести на екран індекси таких елементів. Початковий стан матриці сформувати за допомогою операторів присвоювання.
- Написати програму, яка перевіряє, чи є квадратна матриця розміром $N \times N$ магічним квадратом. Магічним квадратом називається матриця сума елементів кожного рядка, кожного стовпчика і кожної діагоналі є однаковою. Початковий стан матриці сформувати за допомогою типізованої константи-матриці. Далі наведені приклади двох магічних квадратів.

2	9	4
7	5	3
6	1	8

13	8	12	1
2	11	7	14
3	10	6	15
16	5	9	4

- Написати програму, яка перевіряє, чи виконується у квадратній матриці розміром $N \times N$ така умова: сума елементів рядка з номером i дорівнює сумі елементів стовпчика з тим саме номером. Початковий стан матриці сформувати за допомогою операторів присвоювання. Далі наведений приклад матриці у якій виконується вказана умова.

1	4	3	2
3	2	4	5
4	5	6	3
2	3	5	7

- Написати програму, яка перевіряє, чи є квадратна цілочислова матриця розміром $N \times N$

симетричною відносно своїх головних діагоналей. Початковий стан матриці сформувати за допомогою типізованої константи-матриці. Далі наведений приклад матриці у якій виконується вказана умова.

1	2	3	2	1
2	1	2	1	2
3	2	1	2	3
2	1	2	1	2
1	2	3	2	1

19. Написати програму, яка перевіряє, чи є квадратна цілочислова матриця розміром $M \times N$ симетричною відносно свого середнього рядка. Початковий стан матриці сформувати за допомогою типізованої константи-матриці. Далі наведений приклад матриці у якій виконується вказана умова.

1	5	3	4	6	1
1	6	5	4	2	3
5	4	3	2	1	6
1	6	5	4	2	3
1	5	3	4	6	1

20. Написати програму, яка знаходить суму усіх елементів квадратної матриці розміром 10×10 за винятком елементів, що лежать на її головних діагоналях. Елементи матриці – дійсні числа. Початковий стан матриці сформувати за допомогою операторів присвоювання.

Контрольні запитання

1. Чи можуть елементи масиву містити дані різних типів?
2. Якого типу можуть бути елементи масиву?
3. Чи можна змінити в процесі виконання програми кількість елементів масиву?
4. Яким чином можна дістатися конкретного значення елемента масиву?
5. Якого типу можуть бути індекси масиву?
6. Як описати масив? Наведіть приклади.
7. Чи можна використовувати елементи масивів в якості операндів у виразах?
8. В якому випадку можна застосовувати оператор присвоювання для всього масиву?
9. Як здійснюється введення значень масивів з клавіатури?
10. Як здійснюється виведення значень масивів на екран?
11. Як описати масив і ініціювати його з допомогою механізму типізованих констант.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 11

Обробка символічної інформації

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок роботи з символічною (рядковою) інформацією, стандартними процедурами і функціями обробки символічної інформації.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 15](#), уважно розглянути приклади [15.1](#), [15.2](#), [15.3](#) і [15.4](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Кожному символу відповідає число – код символу;
- Отримати доступ до окремого символу рядка можна вказавши його номер у квадратних дужках після імені змінної або константи рядкового типу;
- Максимальна довжина даного рядкового типу складає 255 символів. Цю довжину можна обмежувати при описуванні;
- Рядки можна з'єднувати та порівнювати між собою;
- У випадку, коли задача не розв'язується простою дією, або процедурою, слід розбити її на простіші етапи.

Варіанти завдань

1. Організувати введення з клавіатури будь-якого року в межах від 1 до 3999. Перетворити введене число у римське і вивести на екран.

Далі наведено значення римських чисел:

Одиниці		Десятки		Сотні		Тисячі	
1	I	10	X	100	C	1000	M
2	II	20	XX	200	CC	2000	MM
3	III	30	XXX	300	CCC	3000	MMM
4	IV	40	XL	400	CD		
5	V	50	L	500	D		
6	VI	60	LX	600	DC		
7	VII	70	LXX	700	DCC		
8	VIII	80	LXXX	800	DCCC		
9	IX	90	XC	900	CM		

2. З клавіатури вводиться ім'я файлу і, можливо, через крапку його тип. Перетворити цей запис у такий, що подається в Norton Commander, тобто ім'я розділяється з типом пробілами, тип починає виводитись з 10 позиції, крапка перед ним не ставиться.
3. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення слів на екран з нового рядка, нумеруючи ці слова.
4. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення слів на екран з нового рядка в порядку алфавіту.
5. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення слів на екран у рядок в порядку збільшення їхньої довжини.
6. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення на екран у колонку тих слів, перша літера яких зустрічається в них хоча б ще раз.
7. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення на екран у рядок тих слів, перша літера яких збігається з останньою.
8. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення на екран у колонку тих слів, в яких усі літери різні.
9. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення на екран у рядок тих слів, в яких усі літери однакові.
10. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення на екран у колонку тих слів, які є симетричними (тобто перша літера збігається з останньою, друга – з передостанньою тощо).
11. З клавіатури через пробіл вводиться кілька англійських слів. Організувати виведення на екран у рядок тих слів, які не є симетричними.
12. З клавіатури вводиться послідовність слів. Організувати виведення на екран у рядок цієї послідовності з вилученням зайвих пробілів (тобто залишити між словами по одному пробілу).
13. З клавіатури вводиться послідовність слів. Організувати повторно виведення на екран

цієї послідовності після вилучення зайвих пробілів і поставивши після кожного слова кому а наприкінці рядка – крапку.

14. З клавіатури вводиться послідовність слів. Вивести на екран той символ, який більше всього зустрічається у тексті.
15. Написати програму, яка перевіряє, чи є уведена з клавіатури послідовність символів записом цілого числа. Якщо це так, вивести підтип (*byte*, *integer*, *word* тощо) яким можна визначити це число.
16. Написати програму, яка перевіряє, чи є уведена з клавіатури послідовність символів записом дійсного числа. Якщо це так, вивести підтип (*real*, *double*, *extended* тощо) яким можна визначити це число.
17. Написати програму, яка перевіряє, чи є уведена з клавіатури послідовність символів записом двійкового або шістнадцяткового числа. Якщо це так, вивести відповідне повідомлення і запис числа у десятковій системі числення.
18. З клавіатури або як константа вводиться послідовність крапок і тире, у певних місцях розділені пробілами. Визначити, чи є уведена послідовність азбукою Морзе. Якщо це так перетворити її у текст. Кодування азбуки Морзе наведено нижче:

А .-	Б -...	В .--	Г --.
Д -..	Е .	Ж ...-	З ---..
И ..	Й .----	К -.-	Л .-..
М --	Н -.	О ----	П .--.
Р .-. .	С ...	Т -	У ..-
Ф ..-. .	Х	Ц -.-. .	Ч ----.
Ш ----	Щ --.-	Ъ -...-	Ы -.-.
Ь -...-	Э ..-. .	Ю ..--	Я .-.-

Примітка. За бажанням можна доробити програму, реалізувавши звукову реалізацію уведеної послідовності. У такому випадку слід знати, що частота звучання обирається 500 або 1000Гц, співвідношення тривалості точки і тире становить 1:3, пауза між буквами становить час 4 точок.

19. З клавіатури вводиться речення російською мовою. Перетворити його у запис азбукою Морзе.
Примітка. За бажанням можна доробити програму, реалізувавши звукову реалізацію уведеної послідовності. У такому випадку див. пояснення до завдання 18.
20. Написати програму, яка спочатку буде кодувати уведене з клавіатури текстове повідомлення шляхом збільшення коду символу на 1, виводити на екран кодове повідомлення а після цього дешифрувати повідомлення.

Контрольні запитання

1. Які символи можна використовувати у рядкових змінних?
2. Яким чином можна визначити рядкову змінну?
3. З чого складаються рядкові вирази?
4. Якими способами можна зчепити кілька рядків у один?
5. За якими правилами здійснюється порівняння двох рядків?
6. Призначення процедури *delete*, приклади використання.
7. Призначення процедури *insert*, приклади використання.
8. Призначення процедури *str*, приклади використання.
9. Призначення процедури *val*, приклади використання.
10. Призначення функції *copy*, приклади використання.
11. Призначення функції *length*, приклади використання.
12. Призначення функції *pos*, приклади використання.

[Рекомендації до роботи](#)[Варіанти завдань](#)[Контрольні запитання](#)

Лабораторна робота № 12

Множини

[Рекомендації до роботи](#)[Варіанти завдань](#)[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення і обробки множин, засвоєння основних операцій роботи з ними.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 16](#) і уважно розглянути приклад [16.1](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Елементи множини можуть бути одного зі стандартних скалярних типів даних за винятком дійсного;
- Елементи множини вказуються у квадратних дужках;
- Максимальна кількість елементів множини – 256;
- Над множинами припустимі операції об'єднання (+), перетину (*) и різниці (-). Результатом таких дій є множина;
- Для порівняння двох множин використовують операції порівняння (=, <>, >=, <=, in). Результат порівняння має логічний тип;
- Множинні типи не мають стандартних операторів введення/виведення, програміст сам повинен потурбуватися про такі дії.

Варіанти завдань

1. На підприємстві є 8 цехів (змінна Ceh), які випускають певний асортимент (змінна Asortyment) стандартних кріпильних деталей. Сьогодні кожен цех випускає певний набір таких деталей (оператор Ceh[i] := . . .). Далі наведений початок програми аналізу асортименту стандартних деталей, що випускаються на заводі.

```
Program Pidpryemstvo_1;
```

```
type
```

```
    Product = (gvynt, bolt, gayka, zaklepka, koltso,
               shplint, shurup, shayba, shpilka, shponka, shtift);
```

```
    Asortyment = set of product;
```

```
var
```

```
    Ceh :array[1..8] of Asortyment;
```

```
Begin
```

```
    Ceh[1] := [gvynt, bolt, gayka];
```

```
    Ceh[2] := [gvynt, gayka, zaklepka, shponka];
```

```
    Ceh[3] := [shurup];
```

```
    . . . . .
```

```
End.
```

Визначити повну номенклатуру стандартних кріпильних деталей, яку сьогодні випускалися на підприємстві в цілому.

2. За умовою варіанту 1 визначити номенклатуру стандартних кріпильних деталей, яку сьогодні не випускає жоден цех.

3. За умовою варіанту 1 визначити номенклатуру стандартних кріпильних деталей, яку сьогодні випускали усі цехи підприємства.
4. На підприємстві є 6 цехів (змінна Ceh), які споживають певний асортимент (змінна Asortyment) різального інструменту. Сьогодні кожен цех витратив певний набір такого інструменту (оператор Ceh[i] := . . .). Далі наведений початок програми аналізу асортименту різального інструменту, що споживається на заводі.

```
Program Pidpryemstvo_2;
```

```
type
```

```
    Instrument = (rezetc, freza, sverdlo, zenker, zenkivka,
                 rozvertka, cekivka, mitchyk, plashka, protyagka, shever)
    Asortyment = set of Instrument;
```

```
Ceh :array[1..6] of Asortyment;
```

```
Begin
```

```
    Ceh[1] := [freza, sverdlo, mitchyk, protyagka];
    Ceh[2] := [cekivka, sverdlo, freza];
    Ceh[3] := [mitchyk, sverdlo, zenker, rozvertka];
```

```
    . . .
```

```
End.
```

Визначити повну номенклатуру інструменту, який було витрачено на підприємстві в цілому.

5. За умовою варіанту 4 визначити номенклатуру інструменту, який сьогодні не було витрачено.
6. За умовою варіанту 4 визначити номенклатуру інструменту, який сьогодні витратили усі цехи.
7. В групі (змінна Group) є 25 студентів з прізвищами Name. Шість днів протягом тижня фіксується, хто з студентів був відсутнім (результат присвоюється змінній Vidsutni [. . .]). Далі наведений початок програми аналізу відвідування студентами занять.

```
Program Propuski;
```

```
type
```

```
    WorkDay =
    (Ponedilok, Vivtorok, Sereda, Chetver, Pyatnytsa, Sybota);
    Name = (Bogdanov, Borisenko, Voloshyn, Vasilenko, Danilov,
           Denisenko, Gdanov, Zelinskiy, Zaporojetc, Ivanov,
           Karpenko, Kovalenko, Kolomietc, Kravchenko, Kyrpa,
           Lebedyev, Lemeshko, Maslova, Makarov, Naumenko,
           Ostapenko, Pavlenko, Petrov, Stepanov, Tupolev);
    Group = set of Name;
```

```
var
```

```
    Vidsutni: array[WorkDay] of Group;
```

```
Begin
```

```
    Vidsutni[Ponedilok] := [Denisenko, Kolomietc, Naumenko, Tupolev];
    Vidsutni[Vivtorok] :=
    [Karpenko, Kolomietc, Makarov, Naumenko, Stepanov];
```

```
    . . .
```

```
End.
```

Визначити студентів, які пропустили хоча б один день занять.

8. За умовою варіанту 7 визначити студентів, які жодного разу на цьому тижні не були присутніми.
9. За умовою варіанту 7 визначити студентів, які не мають пропусків на цьому тижні.
10. Студенти (змінна Students) однієї групи можуть отримувати стипендію (Stipendiya), премію (Premiya) та матеріальну допомогу (Dopomoga). Перелік

тих, хто отримував щось цього місяця формується операторами присвоювання (Stipendiya:=...). Далі наведений початок програми аналізу виплат студентами.

Program Vyplaty;

type

```
Name = (Bogdanov, Borisenko, Voloshyn, Vasilenko, Danilov,
        Denisenko, Gdanov, Zelinskiy, Zaporojetc, Ivanov,
        Karpenko, Kovalenko, Kolomietc, Kravchenko, Kyrpa,
        Lebedyev, Lemeshko, Maslova, Makarov, Naumenko,
        Ostapenko, Pavlenko, Petrov, Stepanov, Tupolev);
```

```
Students = set of Name;
```

var

```
Stipendiya: Students;
```

```
Premiya : Students;
```

```
Dopomoga : Students;
```

Begin

```
Stipendiya:= [Borisenko, Voloshyn, Vasilenko, Danilov,
              Denisenko, Zelinskiy, Zaporojetc, Ivanov,
              Karpenko, Kolomietc, Kravchenko, Kyrpa,
              Lebedyev, Maslova, Makarov, Naumenko,
              Ostapenko, Pavlenko, Stepanov, Tupolev];
```

. . .

End.

Визначити студентів, які отримали усі види виплат.

11. За умовою варіанту 10 визначити студентів, які не отримали нічого.
12. За умовою варіанту 10 визначити студентів, які отримували хоча б один з видів виплат.
13. За умовою варіанту 10 визначити студентів, які отримали стипендію і премію.
14. За умовою варіанту 10 визначити студентів, які отримали премію або допомогу.
15. За умовою варіанту 10 визначити студентів, які отримали премію але не отримали стипендію.
16. Є група знайомих людей, які можуть завітати один до одного у гості (змінна Gosti). У масиві Visit зберігаються відомості хто кого відвідував. Далі наведений початок програми аналізу відвідувань.

Program Vidviduvannya;

type

```
Name = (Vasiliy, Volodya, Viktor, Dmytro, Ivan, Inna, Irina, Kostya,
        Lida, Mykola, Natasha, Olena, Olga, Petro, Tetyana, Julia);
```

```
Gosti = set of Name;
```

var

```
Visit : array [Name] of Gosti;
```

Begin

```
Visit[Vasiliy] := [Viktor, Natasha, Petro, Ivan];
```

```
Visit[Volodya] := [Vasiliy, Viktor, Inna];
```

. . .

End.

Визначити, хто з людей не ходив у гості взагалі.

17. За умовою варіанту 16 визначити, хто ходив у гості до всіх.
18. За умовою варіанту 16 визначити, хто хоча б один раз ходив у гості.
19. Студенти (Name) однієї групи для занять спортом можуть обирати одну зі спортивних секцій (Section). Далі наведений початок програми.

Program Sport;

type

```
Name = (Bogdanov, Borisenko, Voloshyn, Vasilenko, Danilov,
        Denisenko, Gdanov, Zelinskiy, Zaporojetc, Ivanov,
```

```

Karpenko, Kovalenko, Kolomiets, Kravchenko, Kyrpa,
Lebedyev, Lemeshko, Maslova, Makarov, Naumenko,
Ostapenko, Pavlenko, Petrov, Stepanov, Tupolev);
Section = (Boxing, Volleyball, Basketball, Gymnastics, Wrestling,
Football, Swimming, Hockey);

```

var

```

Vybor:array[Name] of Section;

```

Begin

```

Vybor[Bogdanov] := Boxing;
Vybor[Borisenko] := football;

```

. . .

End.

Знайти усі секції, які відвідують студенти даної групи.

20. За умовою варіанту 19 вибрати секції у які студенти не записалися.

Контрольні запитання

1. Якого типу можуть бути елементи множини?
2. Скільки елементів може мати множина?
3. Поясніть прикладами порівняння двох множин. Якого типу буде результат таких операцій?
4. Поясніть прикладами операції з множинами. Якого типу буде результат таких операцій?
5. Поясніть переваги і недоліки використання множин у порівнянні з операторами If і Case.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 13

Записи

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок роботи із записами.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 17](#) і уважно розглянути приклад [17.1](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Тип „запис” призначений для впорядкованого збереження інформації різного типу і найчастіше використовується для реалізації баз даних;
- При описування полів у записі тип кожного поля повинен бути описаним окремо;
- При визначенні полів типу string слід ретельно планувати потрібну кількість символів, інакше це призведе до невинного збільшення об'єму бази даних;
- Використання оператора with дозволяє суттєво скоротити і підвищити наочність операторів програми, у якій використовуються дані типу „запис”;
- Pascal дозволяє використовувати записи з варіантами.

Варіанти завдань

1. Створити запис, у якому буде міститися інформація про студента. Відомості повинні включати: Прізвище студента, Код групи (МТ-_1, МТ-_2, МТ-_3), Рік народження (1970..1990), Стать (Чол, Жін), Екзаменаційні оцінки першої сесії, Екзаменаційні оцінки другої сесії.
Створити масив із 15 записів (3 групи по 5 студентів).
Скласти програму, яка вибирає усіх відмінників окремо першої і окремо другої сесії.
2. За умовою варіанту 1 скласти програму, яка вибирає усіх студентів, які стали відмінниками і у першій і у другій сесії.
3. За умовою варіанту 1 скласти програму, яка вибирає усіх студентів, які отримали тільки 4 і 5 у першу сесію.
4. За умовою варіанту 1 скласти програму, яка вибирає усіх студентів, які мають двійки у другу сесію.
5. За умовою варіанту 1 скласти програму, яка вибирає усіх студентів, які вчаться виключно на трійки.
6. За умовою варіанту 1 скласти програму, яка вибирає усіх студентів, які за результатами двох сесій мають середній бал не нижче ніж 4,5.
7. За умовою варіанту 1 скласти програму, яка вибирає усіх студентів, які за результатами першої сесії мають середній бал нижче ніж 3,5.
8. За умовою варіанту 1 скласти програму, яка визначає предмет, який всі групи разом здали найкраще у першу сесію.
9. За умовою варіанту 1 скласти програму, яка визначає предмет, який кожна група окремо здала найкраще у другу сесію.
10. За умовою варіанту 1 скласти програму, яка визначає кількість двійок по кожному предмету за першу сесію.
11. За умовою варіанту 1 скласти програму, яка визначає кількість п'ятірок по кожному предмету за другу сесію.
12. За умовою варіанту 1 скласти програму, яка визначає середній бал по кожному предмету за дві сесії.
13. За умовою варіанту 1 скласти програму, яка визначає предмет, який всі групи разом здали найгірше у другу сесію.
14. За умовою варіанту 1 скласти програму, яка визначає предмет, який кожна група окремо здала найгірше у першу сесію.
15. За умовою варіанту 1 скласти програму, яка визначає предмет, який кожна група окремо здала без двійок у другу сесію.
16. За умовою варіанту 1 скласти програму, яка визначає і виводить на екран список усіх чоловіків віком від 17 до 20 років.
17. За умовою варіанту 1 скласти програму, яка визначає і виводить на екран список усіх дівчат груп МТ-_1 і МТ-_2.
18. За умовою варіанту 1 скласти програму, яка визначає, яка група здала першу екзаменаційну сесію найкраще.
19. За умовою варіанту 1 скласти програму, яка визначає, яка група за результатами двох сесій має найгірші результати.
20. За умовою варіанту 1 скласти програму, яка визначає, яка група має найбільшу кількість відмінників, а яка має найбільшу кількість незадовільних оцінок за результатами двох сесій.

Контрольні запитання

1. Вкажіть призначення даних типу „запис”.
2. Яким чином описати запис?
3. Як можна звернутися до значення поля запису?
4. Чому слід обмежувати довжину полів типу `string`?

5. Призначення і особливості використання оператора `with`.
6. Чи припускається у записі змінна структура полів?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 14

Файли

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок роботи з текстовим і типізованими файлами, засвоєння основних операцій і процедур роботи із файлами.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 18](#), уважно розглянути приклади [18.1](#), [18.2](#), [18.3](#) і [18.4](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Основне призначення файлів (будь-якого типу) – довгострокове збереження інформації і подальша її обробка (друкування, копіювання, знищення тощо) іншими програмами і операційною системою;
- Елементом файла може бути який-завгодно тип Pascal за винятком файлового;
- У програмі, яка звертається до файла (для записування або зчитування інформації) повинна бути описана файлова змінна;
- При описуванні файлової змінної потрібно вибрати один з припустимих типів файла, а саме: текстовий файл (`Змінна: text;`), типізований файл (`Змінна: file of Тип;`) або нетипізований файл (`Змінна: file;`);
- Для того щоб встановити зв'язок між файловою змінною (яка бере участь в операціях з файлами у програмі) і конкретним файлом на зовнішньому носії інформації (на якому фізично зберігається інформація) слід скористатися стандартною процедурою `Assign`;
- При вказуванні імені файла на диску у процедурі `Assign` слід враховувати обмеження операційної системи та компілятора Pascal з якими ви працюєте;
- Після встановлення зв'язку файл слід відкрити для зчитування (процедура `reset`), записування (процедура `rewrite`) або для дописування (процедура `append`);
- Зчитування з файла відбувається за допомогою процедури `read(readln)` а записування за допомогою процедури `write(writeln)`, причому першим параметром таких операторів повинна бути файлова змінна;
- Після завершення усіх операцій з файлом його слід закрити (процедура `close`);
- Робота з файлами – джерело помилок під час роботи програми (відсутній файл з якого слід прочитати інформацію, відсутня або пошкоджена дискета тощо). Потурбуватися про стабільну роботу програми повинен програміст.

Варіанти завдань

Написати **дві** програми.

Перша програма повинна формувати початкові відомості про студентів відповідно до структури, визначеної у лабораторній роботі № 13 і записувати їх у типізований файл.

Друга програма повинна зчитувати інформацію з типізованого файлу (створеного першою програмою), обробляти її у відповідності до індивідуального варіанту завдання до лабораторної роботи № 13, а результати записувати у текстовий файл.

Контрольні запитання

1. Для чого у Pascal реалізовано можливість створення і використання файлів?
2. Які типи можуть бути компонентами файлу?
3. Вкажіть правила запису імен файлів.
4. Як визначити файловий тип у Pascal?
5. Як встановити зв'язок між файловою змінною і конкретним файлом на диску?
6. Як відкрити файл для зчитування, записування або дописування?
7. Як уникати помилок при роботі з файлами?
8. Як відбувається записування і зчитування інформації?
9. Як примусово здійснювати переміщення вздовж файла?
10. Які процедури і функції роботи з файлами на рівні операційної системи ви знаєте?
11. Які особливості роботи з текстовими файлами ви знаєте?
12. Які особливості роботи з типізованими файлами ви знаєте?
13. Які особливості роботи з нетипізованими файлами ви знаєте?

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Лабораторна робота № 15

Модулі

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)

Мета роботи: здобуття практичних навичок створення і використання модулів користувача.

Рекомендації щодо виконання роботи

Перед виконанням роботи слід вивчити [тему 19](#) і уважно розглянути приклад [19.1](#). При вирішенні цієї лабораторної роботи слід пам'ятати про таке:

- Основним призначенням модуля є впорядковане збереження нестандартних типів даних, констант, змінних, процедур і функцій з метою багаторазового використання іншими програмами;
- Модуль має зовнішню секцію, у якій описуються усі елементи, які можна буде використовувати іншими програмами (ця секція починається словом `interface`);
- Модуль має секцію реалізації, у якій реалізуються усі процедури і функції, які були описані у зовнішній секції, а також усі елементи, які потрібні для реалізації перших, але не будуть доступними іншим програмам (ця секція починається словом `implementation`);
- Модуль має секцію ініціювання, яка завжди виконується перед програмою, яка цей модуль використовує, але найчастіше ця секція залишається пустою;
- Модуль можна відкомпілювати (при цьому ви отримаєте файл з розширенням `*.tpu`) але не можна виконати;
- Для того щоб у програмі отримати доступ до елементів модуля слід вказати ім'я цього модуля у розділі `uses`.

Варіанти завдань

Створити модуль і програму, яка буде тестувати модуль.

Модуль повинен складатися з процедур і функцій, які були створені у лабораторних роботах № 8 і 9, типів, які були визначені у лабораторних роботах № 12 і 13.

Тестова програма повинна визначити змінні новостворених типів, а також використати процедури і функції з модуля.

Контрольні запитання

1. Для чого у Pascal реалізовано механізм модулів?
2. Як у програмі отримати доступ до елементів модуля?
3. Яку структуру має модуль Pascal?
4. Що і як слід описувати у зовнішній секції модуля?
5. Що і як слід описувати в секції реалізації модуля?
6. Перелічіть основні стандартні модулі Pascal і поясніть для чого вони призначені.

[Рекомендації до роботи](#)

[Варіанти завдань](#)

[Контрольні запитання](#)