**Тема: “** Команди введення та виведення інформації у мові програмування Паскаль**”**

**Дата:**

**Мета уроку:**

* допомогти учням засвоїти функції введення та виведення у мові програмування Pascal;
* виховання інформаційної культури учнів, уважності, акуратності, дисциплінованості;
* розвинути алгоритмічне мислення, пізнавальні інтереси, навики роботи на комп'ютері.

**Обладнання:**

* комп'ютер, інтерактивна дошка.

**План уроку:**

1. Організаційний момент.
2. Актуалізація знань.
3. Теоретична частина.
4. Практична частина.
5. Домашнє завдання
6. Запитання учнів.
7. Підсумок уроку

**Хід уроку:  
I. Організаційний момент.**

Привітання, перевірка присутніх. Пояснення ходу уроку.

**II. Актуалізація знань.**

1. Що таке алгоритм?
2. Типи алгоритмів.
3. Що таке програма?
4. Структура програми, написаної мовою програмування Паскаль.
5. Змінна, стала, типи змінних.
6. Арифметичні операції.
7. Порядок виконання операцій.
8. Команда присвоювання.
9. Стандартні математичні функції.
10. Що таке модель?
11. Причини моделювання.
12. Ознаки класифікації моделей.
13. Інформаційна модель.
14. Етапи моделювання.
15. Математична модель.
16. Етапи розв’язання прикладних задач з використанням комп’ютера.

**III. Теоретична частина.**

Введення даних - це передача інформації від зовнішнього носія в оперативну пам'ять для обробки.

Виведення - це зворотній процес, коли дані передаються після обробки з оперативної пам'яті на зовнішній носій.

У мові Паскаль стандартним засобом спілкування людини з ЕОМ є консоль, що складається з таких пристроїв, як клавіатура та екран монітора.

***Процедура читання*** **Read** забезпечує введення числових даних, символів, рядків і т.д. для наступної їхньої обробки.

Формат введення:

Read (x1,x2,...xn);

де x1, x2,... xn - змінні допустимих типів даних,

Значення x1, x2,...xn набираються мінімум через один проміжок на клавіатурі, при цьому вони відображаються на екрані для контролю правильності введення. Після набору даних однієї процедури **Read** натискається клавіша **Enter**. Значення введених даних повинні строго відповідати типам цих даних, інакше компілятор виводить на екран повідомлення про помилку.

Якщо в програмі є декілька процедур **Read**, дані для них вводяться потоком, тобто після зчитування даних для однієї процедури **Read** дані для наступної процедури **Read** набираються в тому ж рядку до закінчення рядка.

Процедура читання **Readln** аналогічна процедурі **Read**, єдина відзнака полягає в тому, що після зчитування останнього в списку значення для однієї процедури **Readln** дані для наступної процедури **Readln** будуть зчитуватися з початку нового рядка.

***Процедура запису* Write** робить виведення числових даних, символів, рядків і булівських значень на екран монітору.

Формат виведення:

write (Y1, Y2,... Yn) - виведення на монітор;

де Y1, Y2,... Yn - дані, що виводяться.

Процедура запису **Writeln** аналогічна процедурі **Write,** але після виведення останнього в списку значення для поточної процедури **Writel**n відбувається переміщення курсору до початку наступного рядка.

Процедура **Writeln**, записана без параметрів, виконує тільки переміщення курсору на початок наступного рядка.

Приклад:

Writeln('Y1=',Y1)

У процедурах виведення **Write** **і Writeln** є можливість запису виразу, що визначає ширину поля для виведення. У прикладах, що подаються далі, використовуються наступні умовні позначення:

**І** - ідентифікатор, що визначає ціле число,

**R** - ідентифікатор, що визначає дійсне число,

**р,q** - цілі константи, що визначають довжину поля виведення та кількість цифр після десяткової крапки.

1. Виведення десяткового значення цілої величини **I** у крайні праворуч позиції поля шириною **р** (проміжки перед числом вказані знаком "\_").

Приклади:

Значення **I** Вираз Результат

134 write(I:6); \_\_\_134

1 1 write(I:10,І:10); \_\_\_\_\_\_\_\_\_1 \_\_\_\_\_\_\_\_\_1

312 write(I+I:7); \_\_\_\_624

2. Виведення у крайні праворуч позиції поля шириною **р** значення дійсної величини **R** у форматі з фіксованою крапкою, причому після десяткової крапки виводиться **q** цифр **(0<=q<=24**), що представляють дробову частину числа. Якщо **q=0**, ні дробова частина, ні десяткова крапка не виводяться. Якщо **q>24**, то при виведенні використовується формат із плаваючою крапкою. Без використання форматування число також виводиться у форматі з плаваючою крапкою.

Приклади:

Значення **R** Вираз Результат

511.04 write(R:8:4); 511.0400

-46.78 write(R:7:2); \_-46.78

-46.78 write(R:9:4); \_-46.7800

**Задача** . З клавіатури задається число К. Викинути із запису числа цифру, що позначає сотні. Вивести отримане число на екран.

Program chuslo;  
Uses   Crt;  
Var  
  x, y, k : integer;  
Begin  
  ClrScr;  
  Write (‘Введіть число k=>‘);  
  ReadLn(k);  
  x := k div 1000;  
  x := x \* 100;   
  y := k mod 100;   
  k := x + y;  
  Writeln (k);  
  ReadLn;  
End.

Наприклад:

K=4567;

4567 div 1000 = 4;

4 \* 100 = 400;

4567 mod 100 = 67;

K = 400 + 67 = 467;

**VI. Практична частина.**

**Написати програми для розв’язання наступних задач:**

**Задача 1**. Створити за допомогою програми свою візитну картку.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*                Іванов Сергій                  \*  
\*         Пролетарска 74 кв. 55             \*  
\*              Телефон 445-72-88            \*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Задача 2**. Запросити у користувача два числа і вивести на екран результат суми, різниці, добутку, частки та цілочисельного ділення і залишок від цілочисельного ділення у вигляді таблиці. Наприклад, при введенні чисел 5 і 3 на екрані повинна бути така таблиця:  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*   X   \*   Y   \*   syma \* riznuzya \* dobytok \* chastka \* div   \*   mod    \*   
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*    5    \*    3    \*     8    \*     2     \* 15 \* 1.67 \* 1 \* 2 \*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**V. Домашнє завдання**

1. Вивчити процедури введення та виведення.
2. Написати програми для розв’язання наступних задач:

**Задача 1**. Елементи кола пронумеровані таким чином:

1-радіус (R),

2-діаметр (D),

3-довжина (L),

4-площа круга (S).

Дано радіус кола. Скласти програму і вивести значення інших елементів даного кола (в тому ж порядку).

**Задача 2**. Поміняйте між собою значення двох змінних **А**і **В**, скориставшись третьої змінної **R** для зберігання тимчасового значення.

**VI. Підсумок уроку.**

Підведення підсумку уроку. Виставляння оцінок.