

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

## Зміст

<b>Алгоритми з повторенням і розгалуженням</b>	<b>2</b>
Урок 1. Навчальне програмне середовище АЛГО	2
<i>Вправа 1 «Середовище програмування АЛГО»</i> .....	3
Урок 2. Основні елементи мови Паскаль	4
<i>Вправа 2 «Основні елементи мови Паскаль»</i> .....	6
Урок 3. Складання лінійних програм	6
<i>Вправа 3 «Складання лінійних програм»</i> .....	8
Урок 4. Цілий і логічний типи даних. Оператор розгалуження	8
<i>Вправа 4 «Цілий і логічний типи даних. Умовний оператор»</i> .....	11
Урок 5. Оператор вибору	12
<i>Вправа 5 «Оператор вибору»</i> .....	13
Урок 6. Практична робота 1 «Програми з розгалуженням»	13
Урок 7. Оператори повторення. Цикл із параметром	14
<i>Вправа 7«Оператор повторення. Цикл із параметром»</i> .....	15
Урок 8. Практична робота 2 «Програми з повторенням»	16
Урок 9. Цикл з передумовою	16
<i>Вправа 9 «Цикл з передумовою»</i> .....	18
Урок 10. Цикл з післяумовою	18
<i>Вправа 10 «Цикл з післяумовою»</i> .....	19
Урок 11. Практична робота 3 «Програми з повторенням і розгалуженням»	20

Посібник може використовуватись у 5-7 класах основної школи, має 11 уроків: 8 уроків – теоретичний матеріал із вправою кожного уроку і 3 уроки – практичні роботи. Вправи і практичні роботи розраховані на індивідуальне виконання під час уроку.

Вправи призначені для закріплення теоретичного матеріалу та відпрацювання відповідних практичних навичок поточного уроку. Вправа тривалістю до 25 хв. виконується на кожному уроці. Виконуючи вправу, учень може заробити оцінку 12 балів: 6 балів за знання теорії + 6 балів за виконання практичних завдань.

Знання теорії при виконанні вправ оцінюються за допомогою контрольної-діагностичної системи Test-W2, налаштованої на 6-бальну шкалу. Використовуються тести, підготовлені для 7 класу. Їх можна безкоштовно скачати на сайті <https://aspekt.in.ua/>

Практичні роботи призначені для формування і закріплення теоретичних знань та практичних навичок, здобутих протягом кількох уроків. Для практичної роботи відводиться окремий урок. Виконуючи практичну роботу учень може заробити 12 балів за роботу на компютері, **знання теорії при цьому не перевіряється.**

Для оцінювання практичних навичок у вправах і практичних роботах пропонуються завдання, при виконанні яких учень виконує певні дії, вписуючи, за потреби, відповіді у місця, позначені «\_\_\_\_\_». За кожне завдання, залежно від його складності, можна отримати: до 2 балів, до 3 балів або до 6 балів.

## Алгоритми з повторенням і розгалуженням

### Урок 1. Навчальне програмне середовище АЛГО<sup>1</sup>

У середовищі АЛГО реалізована мова програмування Паскаль, яку розробив Ніклаус Вірт у 1968–1970 роках спеціально для навчання програмуванню і яка одержала визнання завдяки наочності програм і легкості вивчення.

У 1984 році на ринку програмних продуктів з'явилася система програмування Турбо Паскаль фірми Borland, що стало справжньою революцією у програмуванні.

Турбо Паскаль надавав зручне середовище, яке забезпечувало роботу з текстом, відлагодження та запуск програм.

Для запуску розробленої програми необхідна спеціальна програма – компілятор, який перекладає цю програму мовою команд процесора, та текстовий файл, в якому записана сама програма.

Щоб підготувати такий файл, потрібно скористатися текстовим редактором. Крім того, потрібно мати можливість запуску програми на покрокове виконання, мати засоби для роботи з файлами, перегляду проміжних результатів тощо.

Всі ці засоби, зібрані разом, називають системою програмування, а елементи, які відображають на екрані виконувані дії (вікна, меню, кнопки) та забезпечують необхідні сервісні можливості для програміста, – **середовищем програмування**.

Загальний вигляд головного вікна середовища програмування АЛГО наведений на малюнку сторінку).

АЛГО дозволяє відображати команди програми однією з двох мов – англійською або українською. Для переходу на англійську мову, потрібно натиснути кнопку з англійським прапорцем, а на українську – з українським, або натиснувши на клавіатурі клавіші **Ctrl+E** та **Ctrl+U** відповідно.

АЛГО надає можливість автоматичного впорядкування тексту програми (запису кожного оператора з нового рядка, відступів перед вкладеними операторами тощо). Для виконання цієї операції слід вибрати в меню команду **Редагування / Впорядкувати (F4)**.

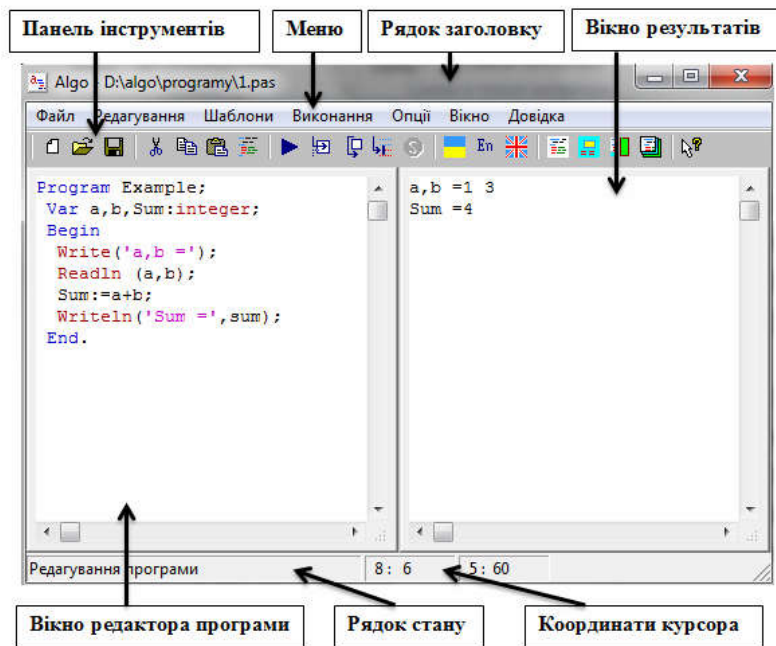
Для більш швидкого набору програми передбачена можливість вставляти в текст оператори, описи та інших конструкції, які можна вибирати з меню **Шаблони**.

Це ж меню можна активізувати клавішею **F10** або натисканням **правої** кнопки миші в режимі редагування програми. Вибраний текст вставляється в програму **на місце положення курсора**. Цим меню користуються також, щоб вибрати колір для команд.

#### Завантаження та збереження програм

Для того, щоб прочитати (завантажити) текст програми з диска, потрібно вибрати команду меню **Файл/Прочитати**. При цьому буде відкрите стандартне вікно для вибору файлу, в якому слід вказати папку та шуканий файл.

Для того, щоб створити новий файл, потрібно вибрати пункт **Файл/Новий** в меню. Якщо поточна програма містить не збережені зміни, то буде виведене відповідне попередження.



<sup>1</sup> АЛГО створив Василь Петрів <http://petriv.ho.com.ua/algo/> м. Львів.

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

Щоб зберегти файл, слід вибрати команду меню **Файл/Записати**. Відкриється стандартне вікно системного діалогу для збереження файлу. Якщо файл вже був записаний і ви тільки вносили зміни, то система автоматично запропонує записати його з тим самим іменем. Зберігаючи новий файл, слід обов'язково ввести з клавіатури його ім'я у відповідному полі діалогу.

Якщо при цьому вказати розширення (.pas, .dat, .txt), то файл буде записаний з цим розширенням, в іншому випадку система автоматично запише файл з розширенням **.pas**.

## **Виконання та відлагодження програми**

Щоб запустити програму на виконання, потрібно вибрати команду меню **Виконання/Виконати (F9)**. Розпочнеться компіляція, яка успішно завершиться лише тоді, коли в тексті програми немає синтаксичних помилок.

Виявивши помилку, компілятор припиняє роботу. Рядок, в якому припинилася компіляція, виділяється червоним кольором і курсор встановлюється на місці зупинки. Можна виправити помилку та знову запустити програму.

Якщо програма скомпільовалася, то розпочнеться її виконання. Проте компілятор не може виявити логічних помилок (наприклад, якщо замість знака додавання написати знак множення). Кнопкою **Stop** або клавішею **F6** завжди можна припинити виконання програми.

Щоб відшукати логічні помилки у програмі, можна виконувати її по одному оператору, для чого вибрати команду **Виконати наступний оператор** або **Трасувати програму** (клавіша **F7**).

## **Питання для комп'ютерного тестування:**

1. Опишіть загальний вигляд головного вікна АЛГО.
2. Прокоментуйте дію основних елементів з панелі інструментів.
3. Як завантажити раніше створену програму?
4. Як зберегти програму?
5. За допомогою яких команд програма запускається на виконання?
6. Які існують режими виконання програми?

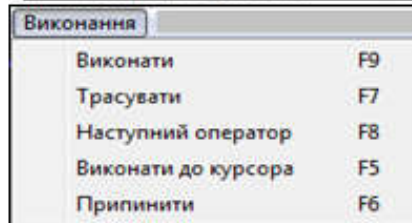
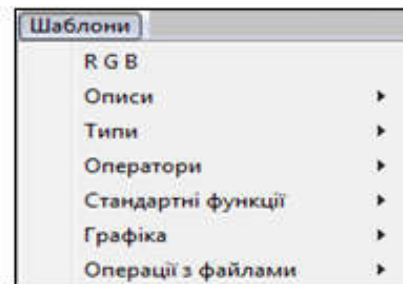
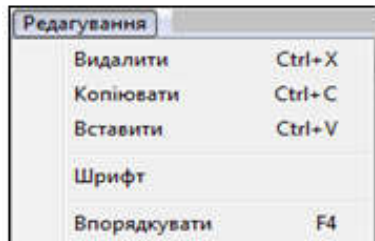
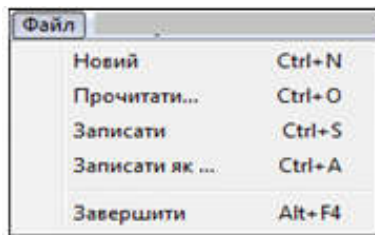
## **Вправа 1 «Середовище програмування АЛГО»**

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7-КЛАС** тест з назвою **УРОК-2-1**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Завантаж програму *Алго*. Відкрий готову програму командою **Файл – Прочитати – tabl\_mnoz**. Запусти програму, відповідай на запитання до одержання оцінки. (до 2 балів)
2. Увімкни українську мову. Переглянь роботу програми у режимі покрокового виконання. Введи у текст програми помилку, спробуй запустити програму. Виправ помилку. (до 2 балів)
3. Вибери команду **Файл – Новий**. Використовуючи меню **Шаблони** і клавіатуру, набери такий текст програми:

```
Program NoName;  
Var i,j : integer;  
Begin  
  For i:=1 to 20 do  
    Begin  
      For j:=1 to 20 do  
        Write( 8 );  
      WriteLn;  
    end  
end.
```

Запусти програму. Закрий програму *Алго*. Набрану програму не зберігай! (до 2 балів)



## Урок 2. Основні елементи мови Паскаль

### Алфавіт і словник мови програмування

При написанні програми використовують знаки, що утворюють алфавіт мови програмування:

- літери англійського алфавіту від **A** до **Z** і від **a** до **z**;
- літери українського алфавіту від **А** до **Я** і від **а** до **я**;
- арабські цифри від **0** до **9**;
- пропуск;
- спеціальні одиничні символи: + - \* / = < > [ ] . , ' ( ) : ; { }
- спеціальні пари символів: <= >= (\* \*) <> .

Неподільні послідовності символів утворюють слова, що несуть у програмі певний зміст. Слова поділяються на зарезервовані та ідентифікатори об'єктів.

**Зарезервовані (службові) слова** є складовою частиною мови, мають фіксоване написання та раз і назавжди визначений зміст.

Наведемо таблицю зарезервованих слів мови Паскаль та їхніх перекладів українською мовою, які використовуються під час роботи АЛГО.

### **Зарезервовані слова мови Паскаль та їхні переклади**

And	та	array	масив
Begin	Початок	case	Вибір
const	Стала	div	Ціла частина
do	виконати	downto	назадДо
else	інакше	end	кінець
for	Для	function	функція
if	Якщо	label	Мітка
mod	Остача	goto	ЙтиДо
or	або	not	не
program	Програма	of	із
repeat	Повторювати	procedure	Процедура
string	Рядок	record	Запис
then	то	type	Тип
until	докиНе	to	до
while	Поки	var	Змінна



**Ідентифікатори (імена)** використовують для позначення типів, констант, змінних, процедур і функцій, які вбудовані у систему програмування або визначені користувачем.

Ідентифікатор може мати довільну довжину, проте до уваги беруться лише перші 16 символів. Він не може починатися з цифри і не повинен містити пропусків. У ідентифікаторах допускаються літери, цифри і знак підкреслення «\_».

Наведемо кілька прикладів ідентифікаторів:

**G, alfa, test17, x2y, \_h1,**

**Сума2Чисел, кінець\_масиву.**

Слова «ім'я» та «ідентифікатор» вживатимемо як синоніми. У АЛГО ідентифікатори стандартних типів, констант, функцій та процедур виділяються в тексті програми червоним кольором.

При написанні зарезервованих слів та ідентифікаторів можна використовувати як великі, так і малі літери.

### Правила оформлення програм

Програма починається із заголовка, що має такий вигляд:

**Program** <ім'я програми>;

**Примітка.** У цьому посібнику кутові дужки <...> означають: «тут слід написати...». У даному випадку замість <ім'я програми> у заголовку пишуть ідентифікатор програми (див. приклади далі).

Після заголовка розміщуються розділи, в яких описуються всі об'єкти (константи, змінні, типи, процедури, функції, мітки), що будуть використані в програмі.

Розділ операторів починається зі службового слова **Begin** і закінчується службовим словом **End** та **крапкою**. У цьому розділі задаються дії над об'єктами програми, оголошеними в розділі описів.

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

Оператори в цьому розділі відокремлюються один від одного крапкою з комою. Так само відокремлюють один від одного розділи програми.

*Приклад*

```
Program Example;  
Var a,b,Sum:integer;  
Begin  
  Write('a,b =');  
  Readln (a,b);  
  Sum:=a+b;  
  Writeln('Sum =',sum);
```

**End.**

Ім'я цієї програми – **Example**. З розділів опису є лише один – розділ опису змінних. Він починається зі службового слова **Var**, після якого записують послідовність оголошень змінних, розділених крапкою з комою.

У кожному оголошенні перераховуються через кому імена змінних одного типу, після чого ставиться двокрапка і вказується тип змінних.

У даному прикладі описано три змінні з ідентифікаторами **a**, **b** та **Sum**, усі вони мають тип **integer**, тобто значення змінних цього типу – цілі числа (*детальніше про типи даних буде далі*).

Після розділу описів змінних іде розділ операторів. Він починається зі службового слова **Begin**, після якого йдуть оператори програми.

**Writeln ('a,b=')** – виклик стандартної процедури для виведення на екран тексту, що міститься між апострофами. На екрані з'явиться текст **a,b=**

Наступний оператор – **Readln(a,b)** – виклик стандартної процедури для читання даних з клавіатури. У даному випадку необхідно ввести два цілих числа ЧЕРЕЗ ПРОПУСК (!), тоді змінна **a** отримає значення, що дорівнює першому введеному числу, а змінна **b** – значення, що дорівнює другому введеному числу.

Наприклад, якщо ввести числа так: **a,b=10 20**, то **a=10**, а **b=20**.

Після цих двох операторів стоїть оператор присвоєння: **Sum:=a+b** (:= – це знак оператора присвоєння).

Під час виконання цього оператора змінна **Sum** набуде значення, що дорівнює сумі чисел **a** і **b**.

Оскільки в результаті додавання двох цілих чисел утворюється ціле число, то змінна **Sum** описана як ціла.

Наступний оператор – це знову оператор виведення **Writeln ('Sum=', Sum)**. Він виведе на екран текст, розміщений між апострофами, а за ним – значення змінної **Sum**. У кінці розділу операторів стоїть службове слово **End**, після якого стоїть крапка.

Якщо в АЛГО перемкнути мову з англійської на українську, то ця програма матиме такий вигляд:

```
Програма Приклад;  
Змінна a,b,Sum:ціла;  
Початок  
  Вивести('a,b=');  
  Ввести (a,b);  
  Sum:=a+b;  
  Вивести('Sum=',sum);  
Кінець.
```



Надалі в більшості прикладів ми будемо використовувати класичний запис програм (англійською мовою).

Для кращого розуміння тексту програм можна користуватись автоматичним перекладом програми, змінюючи мову з англійської на українську кнопками панелі інструментів.

## **Питання для комп'ютерного тестування:**

1. З чого складаються алфавіт і словник мови Паскаль?
2. Що таке зарезервовані слова?
3. Які правила запису ідентифікаторів?
4. З чого починається програма?
5. Як описуються змінні?
6. З чого починається розділ операторів?
7. Як записується оператор виведення?
8. Як записується оператор введення?
9. Як записується оператор присвоєння?
10. Чим закінчується програма?

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

## Вправа 2 «Основні елементи мови Паскаль»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7-КЛАС** тест з назвою **УРОК-2-2**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Завантаж програму *Алго*. Використовуючи меню **Шаблони** і клавіатуру, склади програму для обчислення суми двох чисел за наведеним зразком. У назві програми напиши власне прізвище. Увімкни для тексту програми українську мову. Переглянь роботу програми у режимі покрокового виконання. Праворуч запиши короткі пояснення до кожної команди. Командою **Файл** ⇨ **Записати як...** збережи програму з назвою **Вправа-2А** у власну папку. (до 3 балів)

```
Program Прізвище;
Var
  a,b,Sum : integer;
Begin
  Write( 'a,b =' );
  ReadLn( a,b );
  Sum:=a+b;
  WriteLn( 'Sum =',sum );
end.
```

2. Зміни програму так, щоб обчислювався периметр трикутника за заданими трьома сторонами ( $P=a+b+c$ ). Випробуй програму для різних чисел. Командою **Записати як...** збережи програму з назвою **Вправа-2Б** у власну папку. (до 3 балів)

## **Урок 3. Складання лінійних програм**

У лінійних програмах команди виконуються послідовно, одна за одною.

### Правила запису математичних виразів

Майже в кожній програмі виконуватимуться обчислення, до того ж результати обчислень необхідно буде зберігати для подальшого використання. Для цього існує оператор присвоєння. При його виконанні змінна, ім'я якої стоїть ліворуч від знаку '=', отримує значення виразу, записаного праворуч. Яким би складним не був вираз, він має бути записаний у рядок.

Слід дотримуватися правил запису арифметичних виразів:

- порядок виконання дій змінюють за допомогою круглих дужок. За відсутності дужок пріоритет математичних операцій звичайний: спочатку зліва направо виконуються множення і ділення, потім – додавання і віднімання;
- не можна опускати знак операції множення:  $5ab \Rightarrow 5*a*b$ ;
- звичайні дроби записуються в рядок (зверніть увагу на дужки!):  $\frac{2a-5}{3+b} \Rightarrow (2*a-5)/(3+b)$
- при необхідності у виразах використовуються стандартні функції або функції користувача (див. далі);
- аргументи функцій записуються в круглих дужках.

### Деякі стандартні математичні функції

Функція  $\text{sqr}(x)$  обчислює квадрат значення аргументу  $x$ , тобто  $\text{sqr}(x)=x^2=x*x$ .

Математичний запис	Запис на Паскалі	Значення змінних	Результат
$4^2$	$\text{sqr}(4)$	–	16
$x^2$	$\text{sqr}(x)$	$x=13$	169
$(d+e)^2$	$\text{sqr}(d+e)$	$d=2, e=5$	49
$3^4=(3^2)^2$	$\text{sqr}(\text{sqr}(x))$	$x=3$	81

Функція  $\text{abs}(x)$  обчислює абсолютну величину (модуль) значення аргументу.

Математичний запис	Запис на Паскалі	Значення змінних	Результат
$ 12 $	$\text{abs}(12)$	–	12
$ -12 $	$\text{abs}(-12)$	–	12
$ x+y $	$\text{abs}(x+y)$	$x=3, y=-5$	2
$ x + y $	$\text{abs}(x)+\text{abs}(y)$	$x=3, y=-5$	8

## Стандартні процедури для введення та виведення даних

Більшість програм передбачають введення (наприклад, з клавіатури) користувачем певних даних та виведення результатів роботи (зокрема, на екран монітора).

Мова Паскаль надає всі необхідні засоби для реалізації введення та виведення у програмах.

У попередньому параграфі для цього були використані процедури `Readln` та `Writeln`.

Розглянемо детальніше роботу цих та інших процедур.

**Введення** даних забезпечується викликом процедур `Read` та `Readln`. Якщо передбачається введення кількох значень (наприклад, `Read(a,b,c)`), то їх можна ввести в одному рядку, відділяючи «пропуском», а в кінці натиснути **Enter**.

Можна вводити кожне значення окремо, натискаючи щоразу **Enter**. Змінні одержують свої значення послідовно: спочатку **a**, потім **b** і останньою – **c**.

Введення даних з окремого рядка виконується за допомогою процедури `Readln`.

Для **виведення** повідомлень, значень змінних та виразів використовують процедури `Write` та `Writeln`. Дія процедури `Writeln` відрізняється тим, що після виведення курсор переводиться на новий рядок.

Усі параметри процедури виведення розділяються комами.

*Приклад* при  $a=2, b=3$

```
Writeln(a,b);           на екрані: 2 3
Writeln('a',a);        на екрані: a=2
Writeln('a',a,'b',b);  на екрані: a=2b=3
Writeln('a+b',a+b);    на екрані: a+b=5
```

Для одержання результатів у вигляді таблиць, колонок використовують форматване виведення. При цьому після елемента списку виведення через двокрапку вказується кількість позицій на екрані для виведення її значення.

Якщо позицій більше ніж потрібно, то вони заповнюються пропусками ліворуч від значення.

*Приклад*

```
Writeln(a:5,b:3);      на екрані:      2   3
Writeln('a+b',a+b:4); на екрані: a+b=      5
```

Процедура виведення `Writeln` без параметрів використовується для переведення курсора на новий рядок та виведення порожніх рядків.

### Приклад

Скласти програму для обчислення значення виразу  $y=|x-2|+3$ . Значення змінної  $x$  ввести з клавіатури.

#### *Розв'язування*

```
Program Example;
Var x,y:integer;
Begin
  Write('x=');
  Readln(x);
  y:=abs(sqr(x)-2)+3;
  Writeln('y=',y);
End.
```

### **Питання для комп'ютерного тестування:**

1. Які правила запису арифметичних виразів мовою Паскаль?
2. Запишіть вираз  $y=|x|$  мовою програмування Паскаль.
3. Запишіть вираз  $z=|x-2|+3x^8$  мовою програмування Паскаль.
4. Запишіть вираз  $a=6b^2+|b-3|^3-15$  мовою Паскаль.
5. Що означає команда `Writeln('a,b=')`?
6. Які дії виконуються командою `Readln(a,b)`?
7. За допомогою якої функції обчислити квадрат числа  $x$ ?
8. Які процедури забезпечують введення даних?
9. Які процедури забезпечують виведення даних?
10. Чим відрізняється команда `Writeln` від `Write`?

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

## Вправа 3 «Складання лінійних програм»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7-КЛАС** тест з назвою **УРОК-2-3**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Завантаж програму *Алго*. Склади програму для обчислення виразу  $y=3x^2+12x-5$ , значення  $x$  вводиться з клавіатури. Зразок програми дивись у лівому стовпчику. У правому стовпчику запиши короткі пояснення до кожної команди. Запусти програму для  $x=3$ , запиши результат  $y=$  \_\_\_\_\_ . Збережи програму з назвою **Вправа-3А** у власну папку. (до 3 балів)

```
Program Прізвище;
Var
  x,y : integer;
Begin
  Write( 'x = ' );
  ReadLn( x );
  y:=3*sqr(x)+12*x-5;
  WriteLn( 'y =' ,y );
end.
```

2. Відредагуй програму для обчислення виразу  $z=(a+b)c^2-16$ , значення  $a, b, c$  вводяться з клавіатури. Випробуй програму для  $a=3, b=4, c=7$ , запиши результат  $z=$  \_\_\_\_\_ . Командою **Записати як...** збережи програму з назвою **Вправа-3Б** у власну папку. (до 3 балів)

## Урок 4. Цілий і логічний типи даних. Оператор розгалуження

### Прості типи даних

Щоб описати змінну, необхідно зазначити її тип.

Тип змінної визначає набір значень, яких вона може набувати, форму запису їх в пам'яті та операції, які можуть бути з нею виконані. Типи поділяються на прості та складні.

Змінна простого типу завжди містить один елемент даних (число, літеру і т.п.)

Змінна складного типу являє собою таблицю значень одного типу (масив) або набір полів різних типів (запис).

До простих типів в АЛГО належать:

- цілий тип **integer**;
- символний тип **char**;
- логічний тип **boolean**;
- дійсний тип **real**.

Всі прості типи, крім дійсного, є **порядковими**.



Тип називають **порядковим**, якщо всі його значення можна пронумерувати цілими числами.

### Цілий тип даних

Змінна **цілого** типу може набувати значень з діапазону від  $-2147483648$  до  $2147483647$  і займає в пам'яті 4 байти.

Приклад опису:

```
var a, k, D1, D2 : integer;
```

До даних цілого типу можна застосовувати операції:

«+» – додавання, «-» – віднімання, «\*» – множення, «/» – ділення і деякі інші.

**Примітка.** Оскільки при діленні одного цілого числа на інше не завжди виходить ціле число, то присвоювати результат операції ділення змінній цілого типу **не можна**.

Є дві операції, які застосовують тільки до даних цілого типу і отримують цілочисельний результат: **div** – ціла частина від ділення; **mod** – остача від ділення.

Приклад

19 div 4=4;	19 mod 4=3;
12 div 4=3;	12 mod 4=0;
-21 div 4=-5;	-21 mod 4=-1;
-7 div (-4)=1;	-7 mod (-4)=-3.



## Логічний тип даних

При складанні програм, крім математичних, можна обчислювати значення логічних виразів.

Змінні **логічного** типу можуть набувати тільки одного з двох значень – False (*хибний* або *Ні*) і True (*істинний* або *Так*).

Змінні логічного типу одержують значення в результаті виконання операцій порівняння (відношення):

«<» (менше), «>» (більше), «<=» (менше або дорівнює)  
«>=» (більше або дорівнює), «<>» (не дорівнює), «=» (дорівнює).

Результат операції відношення дорівнює True, якщо відношення задовольняється для значень операндів, що входять у нього, і False – у протилежному випадку.

Приклад опису змінних логічного типу:

```
var m1, m2, dd: boolean;
```

Логічний вираз може бути **простим** (наприклад,  $x > 5$ ) або **складеним**. Складені вирази утворюються з простих за допомогою логічних операцій **and**, **or**, **not** (і, або, не).

Приклад  $(x \geq a) \text{ and } (x \leq b)$ ;  
 $(x > a) \text{ or } (x > b)$ ;  
 $\text{not } (x > a)$ ;

Логічна операція **and** має істинний результат у тому випадку, коли обидва логічні вирази істинні.

Логічна операція **or** має істинний результат у тому випадку, коли істинний хоча б один логічний вираз.

Логічна операція **not** завжди дає результат, протилежний значенню виразу.

Логічні операції, операції відношення й арифметичні операції часто зустрічаються в одному виразі. При цьому відношення, що стоять ліворуч і праворуч від знака логічної операції, потрібно взяти в дужки, оскільки логічні операції мають вищий пріоритет.

Арифметичні та логічні операції мають такий пріоритет:

not, – (унарний)  
and, \*, /, div, mod  
or, xor, +, –  
операції відношення.

Порядок виконання операцій регулюється дужками.

У мові Паскаль не можна вводити логічні дані за допомогою оператора Read. Проте передбачено виведення значень змінних логічного типу за допомогою оператора Write.

Приклад

Обчислити значення виразу:  $(a < b) \text{ and } (b < c) \text{ and } (a < c)$ , при  $a=1$ ,  $b=2$ ,  $c=3$ . Значення виразу дорівнює True, тому що істинними є всі значення простих логічних виразів.

## Оператор розгалуження

Якщо розв'язок задачі має кілька варіантів, що залежать від початкових умов, то при складанні програм використовується оператор розгалуження (або *умовний оператор*).

Він забезпечує виконання чи не виконання команди або блоку команд залежно від заданих умов.

Оператор розгалуження має повну та неповну форми.

Повна форма оператора розгалуження має вигляд:

```
If <умова> Then <команда 1> Else <команда 2>;  
    або  
Якщо <умова> То <команда 1> Інакше <команда 2>;
```

Виконання оператора розгалуження починається з обчислення значення логічного виразу, записаного в умові.

Якщо умова істинна, то виконується <команда 1>, у протилежному випадку – <команда 2>.

Якщо на місці однієї команди потрібно записати кілька, то вони об'єднуються службовими словами **Begin-End**.



## Приклад №1

Вивести на екран більше з двох даних чисел.

### Розв'язування

```
Program Example;  
Var x,y:integer;  
Begin  
  Write ('x,y=');  
  Readln(x,y);  
  If x>y Then Writeln (x)  
    Else Writeln(y);  
End.
```

Зверніть увагу на те, що перед службовим словом Else крапка з комою не ставиться. Алгоритм цієї задачі зображено на малюнку.

**Неповна форма оператора розгалуження** має вигляд:

```
If <умова> Then <команда>;  
  або  
Якщо <умова> То <команда>;
```

Гілка Else може бути відсутньою, якщо при невиконання умови нічого робити не потрібно.

Наприклад, якщо значення змінної – x менше за 0, то замінити його на протилежне. Задача розв'язується за допомогою неповної форми оператора:

```
If x<0 Then x:=-x;
```

Використовуючи оператор розгалуження, слід бути уважним: якщо поставити після Then крапку з комою, програма скомпілюється, але працюватиме неправильно:

```
If x<0 Then; x:=-x;
```

У такому випадку команда x:=-x не є частиною умовного оператора If-Then, і буде виконана обов'язково. Кажуть, що в гілці Then записаний *порожній оператор*.

## Приклад №2

Написати програму для перевірки, чи належить ціле число, введене з клавіатури, інтервалу [0,5].

### Розв'язування

Позначимо через x число, яке вводиться з клавіатури користувачем. За умовою x – це змінна цілого типу. Число x належить заданому інтервалу [0, 5] лише в тому випадку, якщо одночасно виконуються дві умови: (x>=0) і (x<=5). Тому для утворення складної умови скористаємось логічною операцією **and**.

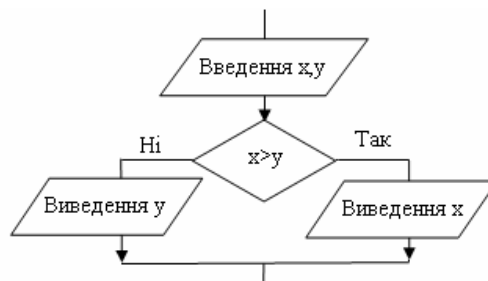
```
Program Example;  
Var x:integer;  
Begin  
  Write ('x=');  
  Readln(x);  
  If (x>=0) and (x<=5)  
    Then Writeln(x, 'належить')  
    Else Writeln(x, 'не належить');  
End.
```

## Вкладені оператори розгалуження

Під час розв'язування задач часто розглядається не два, а більше варіантів. Це можна зробити, використовуючи послідовно кілька умовних операторів. У цьому випадку після службових слів **Then** і **Else** може записуватися новий умовний оператор.

## Приклад №3

Дано цілі числа a,b,c. Якщо  $a \leq b \leq c$ , то всі числа замінити їх квадратами. Якщо  $a > b > c$ , то кожне число замінити найбільшим із них, у інших випадках – змінити знак кожного з чисел.



# «АЛГО (Паскаль) в школі» Ріпко Н.А.

## Розв'язування

Умову задачі перепишемо так:

якщо  $a \leq b \leq c$ , то  $a:=a^2$ ,  $b:=b^2$ ,  $c:=c^2$ ;

якщо  $a > b > c$ , то  $a:=c$ ,  $b:=c$ ,

у решті випадків:  $a:=-a$ ,  $b:=-b$ ,  $c:=-c$ .

**Program** Example;

**Var** a,b,c:integer;

**Begin**

Writeln('Введіть числа a,b,c');

Readln(a,b,c);

If(a<=b) and (b<=c)

Then **begin**

a:=SQR(a); b:=SQR(b); c:=SQR(c)

**end**

Else if (a>b) and (b>c) Then

**begin**

a:=c; b:=c

**end**

Else **begin**

a:=-a; b:=-b; c:=-c

**end;**

Writeln(a:3,b:3,c:3)

**End.**

**Примітка.** Якщо вкладеними умовними операторами є неповні умовні оператори, то можуть виникати проблеми, пов'язані із встановленням меж умовних операторів. У таких випадках службове слово Else відноситься до найближчого If.

## Питання для комп'ютерного тестування:

1. Які типи належать до простих типів даних в АЛГО?
2. Яких значень можуть набувати змінні цілого типу та які операції з ними можна виконувати?
3. Визначити значення логічного виразу:  $(-3 > 5)$  or  $(7 < 9)$  and  $(0 > 3)$ .
4. Записати послідовність операторів для знаходження неповної частки й остачі від ділення цілого числа  $a$  на ціле число  $b$ .
5. Які сполучники використовуються у складних відношеннях?
6. Яким може бути результат логічної операції відношення?
7. Після виконання операторів  
a:=0;  
if a<>0 then; a:=2;  
значення змінної  $a$  дорівнює 2. Поясніть чому.
8. Використовуючи складений оператор, спростіть такий фрагмент програми:  
If a>b then c:=1;  
If a>b then d:=2;  
If a<=b then c:=3;  
If a<=b then d:=4;
9. Яким буде значення змінної  $a$  після виконання операторів:  
a:=3;  
if a<4 then Begin a:=a+2; a:=a+3 End.
10. Запишіть умовний оператор, у якому значення змінної обчислюється за формулою  $a+b$ , якщо  $a$  – непарне,  $a*b$ , якщо  $a$  – парне.

## Вправа 4 «Цілий і логічний типи даних. Умовний оператор»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7-КЛАС** тест з назвою **УРОК-2-4**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Завантаж програму *Алго*. Склади програму для знаходження більшого з двох введених із клавіатури цілих чисел за наведеним зразком. У правому стовпчику запиши короткі пояснення до кожної команди. Запусти програму для таких значень змінних:  $a=1$ ,  $b=3$ ;  $a=3$ ,  $b=1$ ;  $a=1$ ,  $b=1$ . Збережи програму з назвою **Вправа-4А** у власну папку. (до 3 балів)

```

Program Прізвище;
Var a,b:integer;
Begin
  WriteLn('Введіть числа a,b');
  ReadLn(a,b);
  If (a>b) then
    WriteLn(a)
  else
    WriteLn(b);
end.

```

2. Відредагуй програму для знаходження більшого з трьох цілих чисел. Виконай програму для довільних значень змінних. Збережи програму з назвою **Вправа-4Б** у власну папку. (до 3 балів)

### Урок 5. Оператор вибору

Оператор вибору (варіанту) можна розглядати, як узагальнення умовного оператора. Він дає змогу зробити вибір з кількох варіантів залежно від значення змінної.

Виконання оператора вибору починається з обчислення виразу, який повинен мати значення порядкового типу.

Формат запису оператора варіанту такий:

```

Case <порядкова змінна або вираз> of
  <константа 1>:<оператор 1>;
  <константа 2>:<оператор 2>;
  ...
  <константа n>:<оператор n>;
[Else <оператор>; ]
End;

```

або

```

Вибір <порядкова змінна або вираз> is
  <константа 1>:<оператор 1>;
  <константа 2>:<оператор 2>;
  <константа n>:<оператор n>;
[інакше <оператор>; ]
Кінець;

```



У випадку, коли результат обчислення дорівнює одній з перелічених констант, виконується відповідний оператор. Потім керування передається за межі оператора вибору.

Якщо значення виразу не збігається з жодною із констант, то виконується оператор, що стоїть після **Else**, якщо він є, або керування передається оператору, що слідує за **End**.

#### Примітки

1. Тип кожної з констант повинен збігатися з типом виразу. Можна задавати не тільки одну константу, а й список констант (див. приклад).
2. Гілка **Else** міститься у квадратних дужках. Це означає, що ця частина оператора вибору не обов'язкова.
3. У конструкції вибору (на відміну від умовного оператора) перед **Else** ставиться крапка з комою.
4. У якості операторів можуть використовуватися і складені оператори.

#### Приклад

Нехай при тестуванні учень отримав N балів з 20 можливих. Потрібно вивести суму балів з коротким коментарем.

#### Розв'язування

```

Program оцінка;
Var N :integer;
Begin
  Write ('Введіть N-');

```

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

```
ReadLn(N);
Case N of
  20:      WriteLn('Краще не буває!');
  19,18,17 : WriteLn('Відмінно!');
  16,15,14,13:WriteLn('Добре. ');
  12,11,10,9 :WriteLn('Задовільно. ');
  8,7:WriteLn('Ще трохи, і було б добре. ');
else
  WriteLn('Потрібно попрацювати!')
end;
WriteLn('Сума балів - ',N:2,' з 20 можливих');
end.
```

## Питання для комп'ютерного тестування:

1. У яких випадках використовується оператор вибору?
2. Який загальний формат запису оператора вибору?
3. Яких правил потрібно дотримуватися, використовуючи оператор вибору при розв'язуванні задач?
4. Яким повинен бути тип констант?
5. Що означає, коли гілка **Else** міститься у квадратних дужках?
6. Що ставиться перед гілкою **Else**?

## Вправа 5 «Оператор вибору»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7 КЛАС** тест з назвою **УРОК-2-5**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Завантаж програму **Алго**. Склади програму за вказаним зразком. Запусти для різних значень змінної **N**. У правому стовпчику запиши короткі пояснення до кожної команди. Збережи програму з назвою **Вправа-5А** у власну папку. (до 3 балів)

```
Program Прізвище;
Var N :integer;
Begin
Write ('Введіть N - ');
ReadLn(N);
Case N of
  1,2,3,4,5:WriteLn('Робочий день');
  6,7: WriteLn('Вихідний день');
else
  WriteLn('Неправильні дані') ;
end;
end.
```

2. Склади програму для визначення пори року і назви місяця за номером місяця. Якщо введений номер не належить проміжку від 1 до 12 – виведи повідомлення про помилку. Збережи програму з назвою **Вправа-5Б** у власну папку. (до 3 балів)

## **Урок 6. Практична робота 1 «Програми з розгалуженням»**

1. Підкресли оператори, які записані правильно: (до 2 балів)
  - a) if  $x > 0$  then  $y := x - 2$  else  $y := x + 2$ ;
  - b) if  $x = 0$  then  $y := 1024$  else  $y := x - 1024$ ;
  - c) if  $(x = 0)$  or  $(y < 0)$  then  $y := x$ ;
  - d) if  $x = 0$  and  $a > 0$  then  $b := a$  else  $b := x$ ;
2. Запиши мовою Паскаль такі дії: якщо число є парним, то виведи «Так», інакше – виведи «Ні». (до 2 балів) \_\_\_\_\_

3. Запиши мовою Паскаль формулу: (до 2 балів)

$$y = \begin{cases} x, & \text{якщо } x < -5 \\ 2x, & \text{якщо } x \geq -5 \end{cases}$$

4. Підготуй комп'ютер до роботи. Завантаж програму *Алго*. Доповни програму *Вправа-2Б* для обчислення периметра трикутника перевіркою коректності введених даних: у трикутнику сума довжин будь-яких двох сторін повинна бути більшою, ніж довжина третьої сторони. Збережи програму з назвою *Робота-6А* у власну папку. (до 3 балів)
5. Дано число  $N$  ( $N \leq 100$ ), яке позначає вік людини. Додай до цього числа одне зі слів: «рік», «роки», «років» відповідно до норм української мови. Наприклад: 1 рік, 12 років, 52 роки. Збережи програму з назвою *Робота-6Б* у власну папку. (до 3 балів)

### Урок 7. Оператори повторення. Цикл із параметром

При складанні програм часто виникає необхідність багато разів повторити один і той же набір команд. У таких випадках застосовуються *оператори повторення* (циклічні оператори), а команди, що повторюються, називають *тілом циклу*.

Залежно від того, чи відома заздалегідь кількість повторень, розрізняють *цикл з параметром* та *цикли з умовою*.

Оператор циклу з параметром застосовують тоді, коли заздалегідь відоме число повторень певної послідовності операторів.

Для підрахунку кількості повторень вводиться змінна-параметр одного з порядкових типів (integer, boolean, char тощо).

Є дві форми запису циклу з параметром:

- |   |
|---|
| <p>1. <b>For</b> &lt;параметр&gt;:=А to В do &lt;тіло циклу&gt;;<br/> <b>Для</b> &lt;параметр&gt;:=А до В виконати &lt;тіло циклу&gt;;</p> <p>2. <b>For</b> &lt;параметр&gt;:=А downto В do &lt;тіло циклу&gt;;<br/> <b>Для</b> &lt;параметр&gt;:=А назад до В виконати &lt;тіло циклу&gt;;</p> |
|---|

Де А – початкове значення параметра, В – кінцеве значення параметра, тіло циклу – оператор (простий або складений).

Початкове й кінцеве значення параметра циклу можуть бути подані константами, змінними або виразами відповідного типу.

Розглянемо виконання оператора циклу з параметром виду

**For** <параметр>:=А to В do <тіло циклу>;

Спочатку обчислюються значення виразів А і В.

Якщо  $A \leq B$ , то змінна-параметр послідовно набуває значень рівних А, А+1, ..., В-1, В (тобто з кроком 1) і для кожного з цих значень виконується тіло циклу. Якщо на початку  $A > B$ , то тіло циклу не буде виконане жодного разу.

У випадку, коли параметр циклу потрібно зменшувати, використовується друга форма оператора зі службовим словом **downto**. Цикл виконується так само, але значення параметра змінюється з кроком, що дорівнює -1.

Якщо потрібно повторити кілька операторів, то вони об'єднуються службовими словами **Begin-End**.

#### Приклад №1

З чисел від 10 до 99 вивести ті, сума цифр яких дорівнює N ( $0 < N \leq 18$ ).

#### **Розв'язування**

Позначимо через  $k$  чергове число,  $p1$  – старшу цифру числа  $k$ ,  $p2$  – меншу цифру числа  $k$ ,  $S$  – суму.

Число  $k$  будемо друкувати лише в тому випадку, коли сума  $p1$  і  $p2$  дорівнюватиме  $S$ .



# «АЛГО (Паскаль) в школі» Ріпко Н.А.

```
Program Example;  
Var k,N,p1,p2,S:integer;  
Begin  
  Write('N=');  
  Readln (N);  
  For k:=10 to 99 do  
    Begin  
      p1:=k div 10;{виділяємо старшу цифру}  
      p2:=k mod 10;{виділяємо молодшу цифру}  
      S:=p1+p2; {знаходимо суму цифр}  
      If S=N then writeln(k)  
    End  
  End.
```

У цій програмі цикл можна було записати коротше:

```
For k:=10 to 99 do  
  If k div 10+k mod 10=N then writeln(k);
```

Проаналізуйте його роботу самостійно.

## Приклад №2

Знайти всі двоцифрові числа, що діляться на N або містять цифру N.

## **Розв'язування**

Якщо двоцифрове число задовольняє умову задачі, то для нього виконується хоча б одна з трьох умов: перша цифра дорівнює N ( $p1=n$ ) або друга цифра дорівнює N ( $p2=n$ ), або саме число ділиться на N ( $k \bmod n = 0$ ).

*Яку логічну операцію необхідно використати для об'єднання цих простих умов у складену?*

## **Питання для комп'ютерного тестування:**

1. У яких випадках використовуються оператори повторення?
2. Які особливості запису циклу з параметром?
3. До якого типу даних належить змінна параметру циклу?
4. В яких випадках при складанні циклу використовуються службові слова Begin-End?
5. Скільки разів буде виконано тіло циклу у фрагменті програми::  

```
For k:=-1 to 1 do ...
```
6. Скільки разів буде виконано тіло циклу у фрагменті програми::  

```
k:=5; r:=15;  
For i:=k+1 to r-1 do
```
7. Скільки разів буде виконано тіло циклу у фрагменті програми::  

```
k:=5; r:=15;  
For i:=0 to k*r do...
```
8. Скільки разів буде виконано тіло циклу у фрагменті програми::  

```
k:=r;  
For i:=k to r do...
```
9. Визначити значення змінної S після виконання таких операторів:  

```
S:=0; N:=10;  
For i:=2 to N do S:=S+100 div i;
```
10. Перевірте роботу даної програми на комп'ютері. Проаналізуйте використання оператора циклу з параметром:

```
Program Demo;  
Var c:boolean;  
Begin  
  For c:= false to true do  
    writeln(c);  
End.
```

## Вправа 7«Оператор повторення. Цикл із параметром»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму TEST-W2. Відкрий з папки 7-КЛАС тест з назвою УРОК-2-7. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою TEST-W2.

1. Завантаж програму Алго. Склади програму для знаходження суми натуральних чисел з проміжку від 1 до 100, що діляться без остачі на введене з клавіатури число N. Виконай програму. У правому стовпчику запиши короткі пояснення до кожної команди. Збережи програму з назвою **Вправа-7А** у власну папку. (до 3 балів)

## «АЛГО (Паскаль) в школі» Ріпко Н.А.

```
Program Прізвище;
Var k,N,S : integer;
Begin
  Write('N=');
  Readln (N);
  S:=0;
  For k:=1 to 100 do
    If k mod N=0 then S:=S+k;
  Writeln('S=',S);
End.
```

2. Склади програму для знаходження всіх двоцифрових чисел, що діляться на N або містять цифру N (пояснення до задачі див. у підручнику). Збережи програму з назвою **Вправа-7Б** у власну папку. (до 3 балів)

### Урок 8. Практична робота 2 «Програми з повторенням»

1. Підготуй комп'ютер до роботи. Завантаж програму *Алго*. Напиши програму для знаходження всіх дільників натурального числа A, введеного з клавіатури. Запусти програму. Збережи програму з назвою **Робота-8А** у власну папку. (до 3 балів)
2. Доповни програму з п. 1, щоб знаходилась сума дільників введеного числа A. Запусти програму. Збережи програму з назвою **Робота-8Б** у власну папку. (до 3 балів)
3. Доповни програму з п. 1, щоб знаходилась кількість дільників заданого числа A. Запусти програму. Збережи програму з назвою **Робота-8Б** у власну папку. (до 3 балів)
4. Доповни програму для перевірки, чи є введене число A простим. Запусти програму. Збережи програму з назвою **Робота-8** у власну папку. (до 3 балів)

### Урок 9. Цикл з передумовою

У випадку, коли число повторень тіла циклу заздалегідь невідоме, а задається лише умова виконання циклу, використовуються **цикли з умовою**, а саме цикл з передумовою та цикл з післяумовою. У цьому параграфі розглянемо перший з них.

**Оператор циклу з передумовою має вигляд:**

```
While <умова> Do <тіло циклу>;
  або
Поки <умова> Виконати <тіло циклу>;
```

Виконання оператора циклу з передумовою починається з перевірки умови, записаної після слова While. Якщо вона виконується, то виконується тіло циклу, потім знову перевіряється умова і т.д.

Якщо під час чергової перевірки з'ясується, що умова не виконується, то тіло циклу виконуватися не буде. Керування перейде до оператора, що після циклу.

#### Примітки

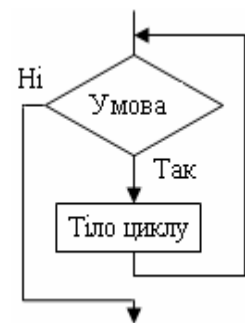
1. Якщо тіло циклу складається з кількох операторів, то вони об'єднуються службовими словами **Begin-End**.
2. У тілі циклу обов'язково має бути оператор, що впливає на істинність умови, інакше станеться зацікловання: оператори тіла циклу будуть повторюватися «вічно».

#### Приклад №1

Підрахувати кількість цифр заданого натурального числа n.

#### Розв'язування

Раніше ми виділяли цифри двоцифрових та трицифрових чисел. У цьому ж випадку ми не знаємо скільки цифр має число.





# «АЛГО (Паскаль) в школі» Ріпко Н.А.

Тому, поки число не стане рівне 0, будемо виконувати таку послідовність команд: збільшувати лічильник кількості цифр числа на одиницю, а число зменшувати в 10 разів (за допомогою цілочисельного ділення позбавлятимемося останньої цифри числа).

```
Program Example;  
Var m, n, k: integer;  
Begin  
  Write('Введіть натуральне число:');  
  Readln(n);  
  m:=n; {копіюємо введене число}  
  k:=0; {змінна-лічильник кількості цифр}  
  While m<>0 Do  
    Begin  
      k:=k+1; {збільшуємо лічильник цифр}  
      m:=m div 10 {відкидаємо останню цифру}  
    End;  
  Writeln('У числі ', n, '-', k, ' цифр')  
End.
```

Роботу цієї програми цікаво спостерігати в покроковому режимі виконання, слідкуючи за зміною значень  $m$  і  $k$ .

## Приклад №2

Підрахувати суму цифр заданого натурального числа  $n$ .

### Розв'язування

Щоб розв'язати цю задачу, досить зробити незначні зміни в попередній програмі. Потрібно, як і раніше, відділяти останню цифру числа, але перед цим її потрібно запам'ятовувати в додаткову змінну (наприклад,  $a$ ) і додавати до суми  $S$ .

```
Фрагмент програми: While m<>0 Do  
  Begin  
    a:=m mod 10;  
    s:=s+a;  
    m:=m div 10  
  End;
```

### Питання для комп'ютерного тестування:

1. У яких випадках використовується цикл із передумовою та які особливості його запису?
2. У даному фрагменті програми обчислення кількості цифр числа  $a$  знайдіть помилку та виправте її.

```
ck:=0;  
While a>=0 Do  
  Begin  
    ck:=ck+1;  
    a:=a div 10  
  End;
```

3. Дано послідовність операторів:

```
a:=1; b:=1;  
While a+b<8 Do  
  Begin a:=a+1; b:=b+2 End;  
  s:=a+b;
```

Скільки разів буде повторюватися тіло циклу? Якими будуть значення змінних  $a$ ,  $b$  і  $s$  після виконання цієї послідовності операторів?

4. Якими будуть значення змінних  $a$  і  $b$  після виконання послідовності операторів:

```
a:=1; b:=1;  
while a<=3 Do a:=a+1; b:=b+1;
```

5. Знайдіть значення змінної  $s$  після виконання таких операторів:

```
a) s:=0; i:=0;  
   While i<5 Do i:=i+1; s:=s+100 div i;  
б) s:=0; i:=0;  
   While i>1 Do
```

```
  Begin s:=s+100 div i; i:=i-1 End;
```

6. Яким умовам повинно задовольняти значення змінної  $k$ , щоб такі цикли були нескінченними:

```
a) While c<0 Do c:=c+k;  
б) While k<>0 Do k:=k+1;  
в) While k<>0 Do k:=k+2;
```



# «АЛГО (Паскаль) в школі» Ріпко Н.А.

## Вправа 9 «Цикл з передумовою»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7-КЛАС** тест з назвою **УРОК-2-9**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Завантаж програму Алго. Склади програму, яка визначатиме кількість парних цифр у натуральному числі N, введеному з клавіатури. Випробуй програму. У правому стовпчику запиши короткі пояснення до кожної команди. Збережи програму з назвою **Вправа-9А** у власну папку. (до 3 балів)

```
Program Прізвище;
Var Ost, N, k: integer;
Begin
  Write ('N=');
  Readln (N);
  k:=0;
  While N<>0 Do
  Begin
    Ost:= N mod 10;
    If Ost mod 2=0 Then k:=k+1;
    N:=N div 10;
  End;
  Writeln ('Парних цифр - ', k)
End.
```

2. Зміни програму так, щоб знаходилась найбільша цифра цілого числа N, введеного з клавіатури. Збережи програму з назвою **Вправа-9Б** у власну папку. (до 3 балів)

## Урок 10. Цикл з післяумовою

Оператор циклу з післяумовою має такий вигляд:

```
Repeat
  <тіло циклу>;
Until <умова зупинки циклу>;
або
```

```
Повторювати
  <тіло циклу>;
ДокиНе <умова зупинки циклу>;
```

Цей оператор відрізняється від циклу з передумовою тим, що перевірка умови проводиться після чергового виконання тіла циклу. **Це забезпечує виконання тіла циклу хоча б один раз.**

Зверніть увагу на те, що даний оператор циклу допускає наявність кількох операторів у тілі циклу, тому службові слова **Begin** і **End** не потрібні.

Порядок виконання циклу з післяумовою такий: виконується послідовність операторів, що складають тіло циклу, після чого перевіряється умова, записана після службового слова **Until**. Якщо умова виконується, то цикл завершується.

У протилежному випадку оператори тіла циклу виконуються ще раз, після чого знову перевіряється виконання умови.

### Приклад

Скласти програму для планування купівлі в магазині товарів на суму, що не перевищує задану величину.

### Розв'язування

Позначимо через  $x$  та  $k$  ціну та кількість товару, через  $p$  – задану граничну суму, через  $s$  – вартість покупки.

Початкове значення загальної вартості покупки  $s$  дорівнює нулю. Значення граничної суми вводиться з клавіатури.

# «АЛГО (Паскаль) в школі» Ріпко Н.А.

Необхідно повторювати запит ціни й кількості вибраного товару, визначати його вартість, додавати її до загальної вартості та виводити результат на екран доти, поки вартість не перевищить граничну суму р.

**Program** Example;

**Var** x, k, p, s: integer;

**Begin**

Write('Гранична сума--');

Readln(P);

S:=0;

Repeat

Write('Введіть ціну товару та кількість:');

Readln(x, k);

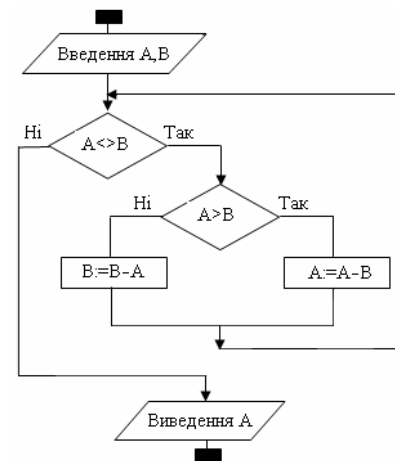
S:=s+k\*x;

Writeln(' вартість покупки дорівнює ', s)

Until s>p;

Writeln('вартість покупки перевищила граничну суму!');

**End.**



## Питання для комп'ютерного тестування:

1. У яких випадках використовується цикл з післяумовою та які особливості його запису?
2. У чому подібність і відмінність циклів з умовами?
3. Виберіть правильну відповідь.

При виконанні оператора повторення *Repeat*

- а) спочатку обчислюється логічний вираз і, в залежності від результату, виконуються або не виконуються оператори тіла циклу;
- б) спочатку виконуються оператори тіла циклу, потім обчислюється логічний вираз, результат якого впливає на повторне виконання операторів.

4. Визначити значення змінної s після виконання таких операторів:

```
s:=0; i:=1;
```

```
Repeat
```

```
  s:=s+5 div i;
```

```
  i:=i-1
```

```
Until i<=1;
```

5. Що буде надруковано в результаті виконання такої послідовності операторів:

```
i:=1;
```

```
repeat
```

```
  write(i, ' ');
```

```
  i:=i+2
```

```
until i>19;
```

6. Визначити значення змінних s та i після виконання таких операторів:

```
S:=0; i:=1;
```

```
repeat
```

```
  S:=S+i;
```

```
  i:=i+1
```

```
until i>10;
```

## Вправа 10 «Цикл з післяумовою»

Займи робоче місце. Підготуй комп'ютер до роботи. Завантаж програму **TEST-W2**. Відкрий з папки **7-КЛАС** тест з назвою **УРОК-2-10**. Вкажи прізвище, ім'я та клас. Відповідай на питання тесту до одержання оцінки. Повідом оцінку вчителю. Заверши роботу з програмою **TEST-W2**.

1. Для обчислення найбільшого спільного дільника (НСД) двох чисел уже більше 20 століть відомий алгоритм, запропонований давньогрецьким математиком Евклідом – *алгоритм Евкліда*. Ідея алгоритму: числа порівнюють і більше з них замінюють різницею цих чисел. Цю дію повторюють до того часу, поки числа не стануть рівними. Отриманий результат і є шуканим НСД двох чисел. Фрагмент програми для реалізації алгоритму має вигляд:

```
Readln (A,B);
```

```
Repeat
```

```
  If A>B then A:=A-B else B:=B-A;
```

```
until A=B;
```

```
WriteLn('NSD=', A);
```

## «АЛГО (Паскаль) в школі» Ріпко Н.А.

Завантаж програму Алго. Напиши програму, яка знаходить НСД двох чисел та випробуй її для таких значень:

а)  $A=12, B=6$ ; б)  $A=8, B=10$ ; в)  $A=5, B=7$ .

Збережи програму з назвою **Вправа-10А** у власну папку. (до 3 балів)

2. Напиши програму для знаходження НСД трьох цілих чисел, враховуючи, що  $\text{НСД}(A, B, C) = \text{НСД}(\text{НСД}(A, B), C)$ . Запусти програму для таких значень:

а)  $A=12, B=6, C=15$ ; б)  $A=21, B=14, C=35$ .

Збережи програму з назвою **Вправа-10Б** у власну папку. (до 3 балів)

### Урок 11. Практична робота 3 «Програми з повторенням і розгалуженням»

1. Напиши значення змінної S, якого вона набуде після виконання таких операторів. (до 2 балів)

1)  $s:=2; i:=0;$

while  $i<5$  do  $i:=i+1; s:=s+1/i;$  \_\_\_\_\_

2)  $s:=5; i:=1;$

while  $i>1$  do begin  $s:=s+1/i; i:=i-1$  end; \_\_\_\_\_

3)  $s:=1; i:=1;$

while  $i<4$  do begin  $i:=i+1; s:=s*i$  end; \_\_\_\_\_

2. Напиши значення змінної S, якого вона набуде після виконання таких операторів: (до 2 балів)

1)  $s:=4; i:=1;$

repeat  $s:=s+1/i; i:=i-1$  until  $i<=1;$  \_\_\_\_\_

2)  $s:=2; i:=1;$

repeat  $s:=s+1/(i+1); i:=i+3$  until  $i>=5;$  \_\_\_\_\_

3)  $s:=1; i:=1;$

repeat  $i:=i+1; s:=s+i$  until  $i>3;$  \_\_\_\_\_

3. Напиши значення змінної S, якого вона набуде після виконання таких операторів: (до 2 балів)

1)  $s:=0; \text{for } i:=10 \text{ downto } 6 \text{ do } s:=s+1;$  \_\_\_\_\_

2)  $s:=0; \text{for } i:=8 \text{ to } 3 \text{ do } s:=s+1;$  \_\_\_\_\_

3)  $s:=1; \text{for } i:=2 \text{ downto } 10 \text{ do } s:=s+1;$  \_\_\_\_\_

4. Підготуй комп'ютер до роботи. Завантаж програму Алго. Напиши програму для знаходження добутку цифр натурального числа N, введеного з клавіатури. Збережи програму з назвою **Робота-2-11А** у власну папку. (до 3 балів)

5. Напиши програму для перевірки, чи містить натуральне число N цифру A (значення змінних N і A вводяться з клавіатури). Збережи програму з назвою **Робота-2-11Б** у власну папку. (до 3 балів)