**Урок № 9**

**Тема: “** Команда повторення. Оператори циклу мовою Паскаль (WHILE, REPEAT, FOR)**”**

**Мета уроку:**

***навчальна:*** навчити розв’язувати задачі з використанням команди

повторення та записувати їх мовою програмування.

***виховна:*** виховувати у учнів цікавість до предмета, бережливість, почуття відповідальності

***розвиваюча:*** розвивати у учнів логічне мислення, уяву, увагу, пам'ять.

**Тип уроку:** засвоєння нових знань.

**Обладнання:**

* комп'ютер, інтерактивна дошка.

**План уроку:**

1. Організаційний момент.
2. Актуалізація знань.
3. **Вивчення нового матеріалу з поетапним закріпленням**.
4. Домашнє завдання
5. Запитання учнів.
6. Підсумок уроку

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

Привітання, перевірка присутніх. Пояснення ходу уроку.

**II. Актуалізація знань.**

1. Запишіть повну та скорочену форми оператора розгалуження мовою програмування Паскаль. Як потрібно розуміти ці записи?
2. Що таке складений оператор? Як у тексті програми визначити, чи є цей оператор складеним?
3. Що називають вкладеним умовним оператором?
4. Як визначити, до якого з кількох операторів if належить вітка else?
5. Що є умовою в операторі розгалуження? Якими можуть бути умови цього оператора?
6. З чого складається складена умова оператора розгалуження? Які є правила записування складеної умови?
7. Які логічні операції ви знаєте? Від чого залежить результат логічної операції? Чи однаковий пріоритет мають різні логічні операції?
8. Оператор вибору.

# ІІІ. Пояснення нового матеріалу

# Цикли

# Цикл – це багаторазове повторення послідовності дій.

# Дії, які повторюються називаються – тілом циклу.

# Використовують в програмуванні три форми циклів:

# *цикл з параметром*,

# *цикл з передумовою,*

# *цикл з післяумовою* .

# Цикли реалізують за допомогою алгоритмічних конструкцій трьох видів:

# Цикл “для” ( For … to …do)

# Цикл “поки” (while … do )

# Цикл “повторити” (repeat … until)

# Самий універсальний цикл WHILE

# Цикл While називають циклом з *передумовою,* тому що спочатку перевіряється умова циклу, а потім формується тіло циклу.

# Будь – яку циклічну програму можна описати за допомогою циклу While.

# Рассказать хочу я вам

# Про чудесный цикл WHILE,

# На уроке без него

# Не решите ничего.

# Чтобы этот цикл понять

# Надо правила все знать

# Зададим для переменной Х

# Мы начальное значение Х:=1;

# Словом While тем более While x<=14 do

# Зададим условие

# Дальше видим цикла тело Begin

# Без него не будет дела, Writeln(‘шановні друзі, я вас вітаю’);

# Ну, а в теле не забудь Writeln(‘ Рада вас бачити‘);

# Прирост сделать как– нибудь Х:=х+1;

# В цикле обязательны End;

# Скобки операторные,

# Очень важно не забыть

# Эти скобки нам закрыть

# Эти правила нужны

# И знать все мы их должны.

# И тогда цикл While, быть может

# Вам в решении поможет.

# Цикл While для вас теперь

# Вмиг откроет свою дверь,

# И желает вам удач

# В решении любых задач!

# Це дуже просто-

# Вкажіть початкове значення змінної величини;

# Поставте умову циклу Увага!

# While умова do

# Відкрийте операторні дужки

# Begin

# Змонтуйте *тіло циклу*

# (приріст змінної величини, незабудь)

# Закрийте операторні дужки end;

# Вивести результат можна в циклі або за циклом

# Одним із самих найбільш розповсюджених на практиці обчислень алгоритмів циклічної структури являються алгоритми обчислення деякої функції y=f(x) для значень x, яке змінюються від початкового значення x0 до кінцевого xk з кроком h.

Блок – схема

початок

х0, хк, h

x:=x0

x<=xk

y=f(x)

x, y

x:=x+h

кінець

ні так

# Цикл з післяумовою

# Виконується один раз, якщо умова циклу є хибною.

# Блок - схема.

# 

x:=x+0.5

x>=5

y:=2\*x+3

вивести X,Y,

x:=1;

**Початкове значення змінної**

**Repeat**

Тіло циклу

Приріст змінної величини

**Until** умова циклу (цикл працює до тих пір поки умова несправедлива)

uses crt;

var y,x:integer;

begin clrscr;

writeln(funcsia’);

x:=1; {початкове значення змінної величини }

REPEAT {повторити всі команд( які між командами REPEAT і UNTIL}

Y:=2\*x+3; {тіло циклу}

writeln(x:6:1, y:6:1); {тіло циклу команди, що періодично повторюються}

x:=x+0.5; {тіло циклу -- приріст змінної величини}

UNTIL x>5; {умова циклу ( цикл працює до тих пір поки умова несправедлива)}

end.

# Цикл з післяумовою.

# Для циклів з післяумовою треба запам’ятати:

# До слова REPEAT записуються початкове значення змінної величини.

# Дії, які треба повторити записуються між словами REPEAT … UNTIL.

# Якщо умова істинна, то цикл закінчується(умова циклу є умовою виходу із циклу)

# Цикл з післяумовою може виконуватись яку завгодно кількість разів.

# Параметр циклу (змінна величина) може бути довільного типу: INTEGER або REAL.

# Параметр циклу (змінна величина) ми можемо збільшувати або зменшувати на яку завгодно величину (приклад Х:=Х+2; або Х:=Х+0.3 )

# Цикл з післяумовою завжди виконується хоча б один раз, незважаючи на значення змінної величини, що є його параметром.

# Якщо ми забудемо змінювати змінну величину(параметр, приклад Х:=Х+0.2) , то такий цикл буде виконуватися „ вічно”.

# Оператор повтору *for*.

# Цей оператор повторення інакше називається циклом з параметром (або лічильником).

# Він складається із заголовка та тіла циклу та може бути поданий у двох форматах.

# 1. *for* <параметр циклу> := <S1> *to* <S2> *do* *begin* <серія>; *end*;

# 2. *for* <параметр циклу> := <S1> *downto* <S2> *do* *begin* <серія>; *end*;

# де S1 і S2 - вирази, що визначають відповідно початкове і кінцеве значення параметру циклу;

# *for*.. *do* - заголовок циклу; <серія> - тіло циклу.

# Тіло циклу може бути простим або складеним. Цей оператор забезпечує виконання тіла циклу доти, поки не будуть перебрані всі значення параметра циклу від початкового до кінцевого.

# Параметр циклу, його початкове і кінцеве значення повинні належати до одного типу даних.

# При цьому можливо використання будь-якого скалярного типу, крім дійсного, тобто цілого (*integer*, *byte*, *word*, *longint*), булівського (*boolean*) та символьного (*char*). Значення параметра циклу послідовно збільшується при *for... to* або зменшується при *for... downto* на одиницю при кожному повторі.

# Кількість повторів тіла циклу в операторі for можна визначити за наступною таблицею:

|  |  |  |  |
| --- | --- | --- | --- |
| *Оператор* | *S1 < S2* | *S1 = S2* | *S1 > S2* |
| for...to | S2-S1+1 раз | 1 раз | Не виконується |
| for...downto | Не виконується | 1 раз | S1-S2+1 раз |

# В операторі *for* після *do* може знаходитися складений оператор, в тілі якого *заборонені оператори, що змінюють значення параметра циклу*. Після нормального завершення циклу значення параметра циклу дорівнює кінцевому значенню.

# Цикл може не виконатися зовсім (дивись таблицю), але ніколи не може зациклитись на відміну від попередніх двох операторів повторення.

# *Приклад програми з оператором For*:

# *Умова задачі*. Знайти суму всіх натуральних чисел від 1 до 100.

# *Розв'язок*:

**Program** syma;

**Uses** crt;

**Var** Sum, i : integer;

**Begin**

Clrscr;

Sum := 0;

**For** i := 1 **to** 10 **do**

Sum := Sum + i;

Writeln(‘Sum = ‘, Sum);

Readkey;

**End**.

# Порівняльна характеристика циклічних програм.

Для любих чисел, що вводяться з клавіатури обчислити добуток тих чисел, що кратні числу 3.

|  |  |  |
| --- | --- | --- |
| uses crt;  Var n,i,a,d:integer;  {a - chislo}  begin clrscr;  writeln('vvectu n');  readln(n);  d:=1;  **for i:=1 to n do**  **begin**  *writeln('vvestu chislo -a');*  *readln(a);*  *if a mod 3=0 then d:=d\*a;*  **end;**  writeln('d=',d:4);  end | uses crt;  Var n,i,a,d:integer;  {a - chislo}  begin clrscr;  writeln('vvectu n');  readln(n);  d:=1; **i:=1;**  **while i<= n do**  **begin**  *writeln('vvestu chislo -a');*  *readln(a);*  *if a mod 3=0 then d:=d\*a;*  ***i:=i+1;***  **end;**  writeln('d=',d:4);  end. | uses crt;  Var n,i,a,d:integer;  {a - chislo}  begin clrscr;  writeln('vvectu n');  readln(n);  d:=1; **i:=1;**  **repeat**  *writeln('vvestiu chislo -a');*  *readln(a);*  *if a mod 3=0 then d:=d\*a;*  ***i:=i+1;***  **until i>n;**  writeln('d=',d:4);  end. |

**ЗАДАЧА**

Відомо ряд натуральних чисел від 10 до 20. Обчислити суму тих чисел, що кратні числу 4.

**Урок № 10**

**Тема: “** Команда повторення. Оператори циклу мовою Паскаль (WHILE, REPEAT, FOR)**”**

**Дата: 19.03.2014 р.**

**Мета уроку:**

***навчальна:*** навчити розв’язувати задачі з використанням команди

повторення та записувати їх мовою програмування.

***виховна:*** виховувати у учнів цікавість до предмета, бережливість, почуття відповідальності

***розвиваюча:*** розвивати у учнів логічне мислення, уяву, увагу, пам'ять.

**Тип уроку:** практична робота.

**Обладнання:**

* комп'ютер, інтерактивна дошка.

IV. Практична робота №10

# Тема: „ Побудова алгоритмів з циклами та їх реалізація у вигляді програм”

#### Хід роботи

# Завдання практичної роботи

# Скласти програми для розвязку таких задач:

### **Обчислити таблицю значень функції y=sin3x для х від -2 до 4 з кроком 0.5.**

### **Скласти програму, яка виводить таблицю значень функції y=x/cos2x для х від -1 до 2 крок 0.5.**

### **Для любих чисел, що вводяться з клавіатури, обчислити кількість непарних чисел.**

### **Визначити суму квадратів цілих чисел від одиниці до *х* (значення *x>1* задає користувач).**

### **Визначити добуток цілих чисел від *а* до *с* (значення цілих чисел *а* та *с* задає користувач).**

### **Визначити середнє арифметичне натуральних чисел від 10 до у (значення натурального числа у>10 задає користувач).**

### **Для заданого цілого числа *x* обчислити значення функції:**

### **.**

**IV. Підсумок уроку.**

Підведення підсумку уроку.