

зав. рен.
Директор



Отзыв

об уроке информатики в 8 классе по теме «Циклический алгоритм», проведенного учителем информатики МКОУ «Кадиркентская СОШ» Магомедовой Умусалимат Курбановной

Тип урока: Урок открытия новых знаний, обретения новых умений и навыков

Вид урока: комбинированный. В урок включены проверка знаний, работа над пройденным материалом, изложение нового материала, закрепление нового материала, подведение итогов урока и постановка домашнего задания.

На уроке ставились **цели:** активизация познавательной деятельности учащихся, получить учащимся представление о цикле циклическом алгоритме, воспитание информационной культуры учащихся, внимательности, алгоритмического мышления.

На уроке были учтены возрастные и психологические особенности учащихся. Организация урока была на высшем уровне. Учитель и ученики были готовы к началу урока, были соблюдены правила поведения в компьютерном классе и техника безопасности. В конечном итоге работы цели, поставленные учителем, были достигнуты.

Учитель продемонстрировал отличное владение педагогическим мастерством и методикой преподавания. Во время урока речь была понятной и доступной ученикам. Главная информация подчеркивалась.

Используемые на уроке методы объяснения материала и практические методы позволили ученикам воспринять информацию без особых сложностей. Структура проведения урока знакома учащимся, что позволяет не тратить время на лишние вопросы и организационные моменты. Для закрепления нового материала были решены разные примеры. Ученики активно работали на уроке. Внимание учеников привлекала информация, выводимая на слайдах презентации. Главная цель на этом уроке – программирование циклического алгоритма с помощью оператора `repeat` и получение опыта программирования циклов с помощью оператора `repeat` – достигнута, все ученики освоили материал урока.

Выполненные задания позволили учащимся окончательно разобраться в новой теме и закрепить полученные знания. Во время урока именно учение учителя общаться с учащимися помогло достичь всех поставленных целей и создать рабочую атмосферу. Тактичность совмещалась с юмором, строгость с похвалой, это не позволило учащимся забыть о присутствии учителя на уроке.

В целом, урок был отлично спланирован, цели достигнуты, считаю, что урок удался.

Урок посетили: Мусаева З.А., зам. Директора по УВР
Алиева А.З., учитель математики, руководитель ШМО





МКОУ «КАДИРКЕНТСКАЯ СОШ»
СЕРГОКАЛИНСКОГО РАЙОНА РЕСПУБЛИКИ ДАГЕСТАН

Открытый урок

в 8 классе
на тему:

«Циклические алгоритмы»

**Подготовила:
учитель информатики
Магомедова У.К.**

Кадиркент, 2024г.

Предмет: Информатика. Урок ознакомления с новым материалом

Тема: «Циклические алгоритмы»

Продолжительность: 1 урок - 40 минут

Класс: 8 класс

Технологии: Презентация, интерактивная доска

Цель: изучение алгоритмической структуры цикла, создание моделей и алгоритмов для решения практических задач.

Х о д у р о к а

I. Актуализация знаний

- Повторить понятие алгоритма, основные конструкции алгоритмического языка.
- Уметь разрабатывать математическую модель, алгоритм и блок схему решения задачи.
- Иметь понятие о языках программирования и их назначении.
- Уметь работать в среде программирования.
- Знать структуры программы.
- Уметь записывать выражения, содержащие числовые и символьные величины.
- Знать структуры операторов и особенности их работы.
- Уметь применять операторы при написании программ с линейными и ветвящимися структурами.
- Уметь на компьютере создавать и запускать программы на отладку.

II. Теоретический материал урока

Большинство практических задач требует многократного повторения одних и тех же действий, т. е. повторного использования одного или нескольких операторов.

Вспомним времена года. Что мы можем сказать, повторяются ли они ежегодно?

Каждое утро мы совершаем одно и то же – идем в школу, каждый вторник третьим уроком проходит информатика, после уроков идем домой. И это повторяется ежедневно. Кто сможет привести примеры на повторение?

Просмотрим интерактивный материал из библиотеки РИС «ЭОД»

<https://resh.edu.ru/subject/lesson/3062/start/>

Мы просмотрели видеоматериал. О чем мы сегодня будем говорить?

Пусть требуется ввести и обработать последовательность чисел. Если чисел всего пять, можно составить линейный алгоритм. Если их тысяча, записать линейный алгоритм можно, но очень утомительно и нерационально. Если количество чисел к моменту разработки алгоритма неизвестно, то линейный алгоритм принципиально невозможен.

Другой пример. Чтобы найти фамилию человека в списке, надо проверить первую фамилию списка, затем вторую, третью и т.д. до тех пор, пока не будет найдена нужная или не будет достигнут конец списка. Преодолеть подобные трудности можно с помощью циклов.

Циклом называется многократно исполняемый участок алгоритма (программы). Соответственно циклический алгоритм — это алгоритм, содержащий циклы.

Различают два типа циклов: с известным числом повторений и с неизвестным числом повторений. При этом в обоих случаях имеется в виду число повторений на стадии разработки алгоритма.

Существует 3 типа циклических структур:

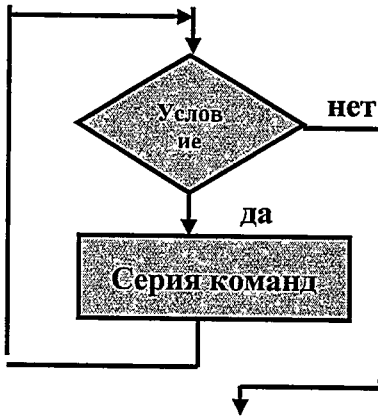
- Цикл с предусловием;
- Цикл с постусловием;
- Цикл с параметром;

Иначе данные структуры называют циклами типа «Пока», «До», «Для».

Графическая форма записи данных алгоритмических структур:

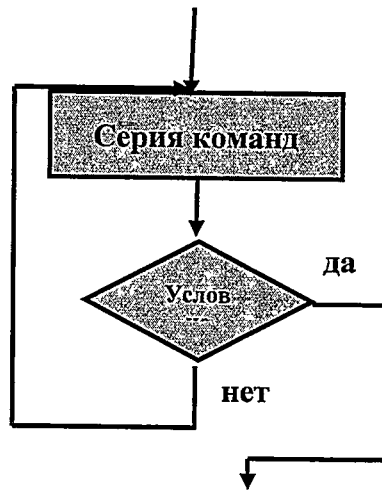
Виды циклических алгоритмов

**Цикл с
предусловием**



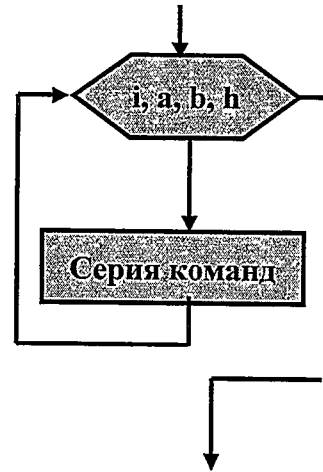
Цикл типа *Пока*

Цикл с постусловием



Цикл типа *ДО*

Цикл с параметром



Цикл типа *ДЛЯ*

Цикл с предусловием (иначе цикл *пока*) имеет вид:

Форматы записи операторов алгоритма	Блок-схема	Форматы записи операторов на Паскале
<u>Пока</u> (условие) <u>нц</u> серия команд <u>кц</u>		while условие do begin серия команд; end;

где

условие – выражение логического типа.

Цикл может не выполняться ни разу, если значение логического выражения сразу же оказывается ложь.

Серия команд, находящихся между **begin** и **end**, выполняются до тех пор, пока условие истинно.

Для того чтобы цикл завершился, необходимо, чтобы последовательность инструкций между **BEGIN** и **END** изменяла значение переменных, входящих в *условие*.

Цикл с постусловием (иначе цикл *до*) имеет вид:

Форматы записи операторов алгоритма	Блок-схема	Форматы записи операторов на Паскале
<i>В алгоритмическом языке нет команды которая могла бы описать данную структуру, но ее можно выразить с помощью других команд (Например, ветвления).</i>		repeat серия команд until условие

где

условие – выражение логического типа.

Обратите внимание:

Последовательность инструкций между **repeat** и **until** всегда будет выполнено хотя бы один раз;

Для того чтобы цикл завершился, необходимо, чтобы последовательность операторов между **repeat** и **until** изменяла значения переменных, входящих в выражение условие.

Инструкция **repeat**, как и инструкция **while**, используется в программе, если надо провести некоторые повторяющиеся вычисления (цикл), однако число повторов заранее не известно и определяется самим ходом вычисления.

Цикл с параметром (иначе цикл для) имеет вид:

Форматы записи операторов алгоритма	Блок-схема	Форматы записи операторов на Паскале
<u>Для i от a до b шаг h</u> <u>делай</u> <u>Нц</u> Серия команд <u>кц</u>		<u>h = +1</u> for <i>i</i> := <i>a</i> to <i>b</i> do begin серия команд end; <u>h = -1</u> for <i>i</i> := <i>b</i> downto <i>a</i> do begin Серия команд; end;

где

i- параметр цикла;

a – начальное значение цикла;

b- конечное значение цикла;

h-шаг изменения параметра.

Структура данного цикла иначе называют циклом i раз.

Эта команда выполняется таким образом: параметру i присваивается начальное значение a, сравнивается с конечным значением b и, если оно меньше или равно конечному значению b, выполняется серия команд. Параметру присваивается значение предыдущего, увеличенного на величину h - шага изменения параметра и вновь сравнивается с конечным значением b.

На языке программирования Паскаль шаг изменения параметра может быть равным одному или минус одному.

Если между begin и end находится только один оператор, то операторные скобки можно не писать. Это правило работает для цикла типа «Пока» и «Для».

Рассмотрим примеры решения задач с использованием данных структур

Примеры

№1

i=1

while i<=10:

print(i)

i=i+1

№2

p=1

i=1

while i<6:

p=p*2

i=i+1

print(p)

№3

i=1

while True:

i=i+1

print(i)

if i > 10: break

№4

s=0

i=1

while True:

s=s+i

i=i+1

if i > 5: break

print(s)

№5 Цикл с параметром

s=0

p=0

for i in range(5):

s = s + p

p = p + 6

print(s)

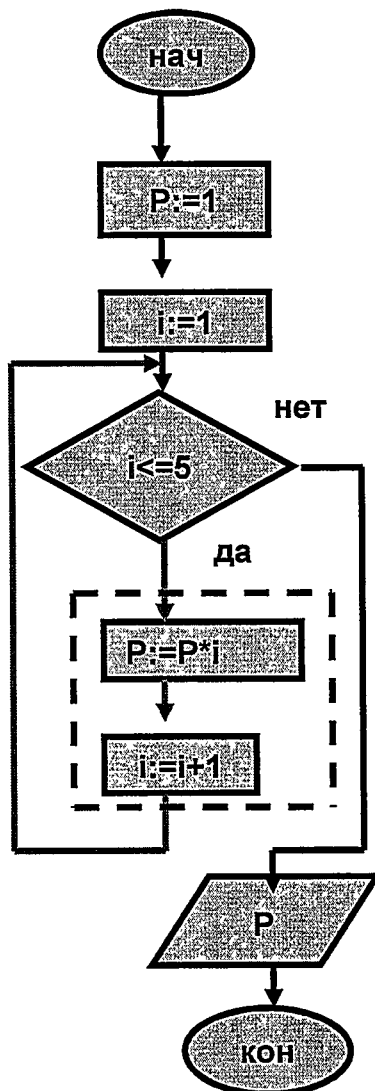
Пример №6

Вычислить произведение чисел от 1 до 5 используя различные варианты цикла

Математическая модель:

$$P = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$$

Составим алгоритм в виде блок-схемы.



Для проверки правильности алгоритма заполним трассировочную таблицу.

Шаг	Операция	P	i	Проверка условия
1	P:=1	1		
2	i:=1;	1	1	
3	i<=5 P:=P*I i:=i+1	1	1	1<=5, да (истина)
4	i<=5 P:=P*I i:=i+1	2	2	2<=5, да (истина)
5	i<=5 P:=P*I i:=i+1	6	3	3<=5, да (истина)

6	$i \leq 5$ $P := P * I$ $i := i + 1$	24	4	$4 \leq 5$, да (истина)
7	$i \leq 5$ $P := P * I$ $i := i + 1$	120	5	$5 \leq 5$, да (истина)
8	$i \leq 5$ $P := P * I$ $i := i + 1$			$6 \leq 5$, нет (ложь)

Проверка условия происходит в несколько шагов: проверка условия и выполнение команд на одной из ветвей. Поэтому в трассировочной таблице записываются не команды алгоритма, а отдельные операции, выполняемые компьютером на каждом шаге.

Шаг первый: Р присваивается значение один.

Шаг второй: i присваивается значение один.

Шаг третий: при i равном единице проверяем условие один меньше или равен пяти, да, условие истинно, значит Р присваивается значение один умноженное на один, будет два. Для i: один плюс один, будет два.

Шаг четвертый: при i равном двум проверяем условие два меньше или равен пяти, да, условие истинно, значит Р присваивается значение 2 умноженное на один, будет 2. Для i: два плюс один, будет три.

Шаг пятый: при i равном трем проверяем условие три меньше или равен пяти, да, условие истинно, значит Р присваивается значение два умноженное на три, будет шесть. Для i: три плюс один, будет четыре.

Шаг шестой: при i равном четырем проверяем условие четыре меньше или равен пяти, да, условие истинно, значит Р присваивается значение шесть умноженное на четыре, будет двадцать четыре. Для i: четыре плюс один, будет пять.

Шаг седьмой: при i равном пяти проверяем условие пять меньше или равен пяти, да, условие истинно, значит Р присваивается значение двадцать четыре умноженное на пять, будет сто двадцать. Для i: пять плюс один, будет шесть.

Шаг восьмой: при i равном шести проверяем условие шесть меньше или равен пяти, нет, условие ложно, тогда мы выходим из цикла, а в результате получаем последнее значение равное сто двадцати.

Program Pr1;

Var i: integer;

Begin

P:=1;

i:=1;

While i<=5 do

begin

P:=P*i;

i:=i+1;

end;

Write ('P=', P);

end.

Для цикла с постусловием построим блок-схему и трассировочную таблицу.

В результате получаем последнее значение равное сто двадцати на седьмом шаге

И для Цикла с параметром построим блок-схему и трассировочную таблицу.

В результате получаем последнее значение равное сто двадцати на шестом шаге

Задача:

Вывести на экран числа от 1 до 5 в:

- а) прямом порядке;
- б) обратном порядке.

Математическая модель:

- а) 1 2 3 4 5;
- б) 5 4 3 2 1.

Блок-схема и программа решения задачи представлена для чисел в прямом порядке и обратном порядке.

Запишем рассмотренные алгоритмы на языке программирования Паскаль.

III. Подведение итогов урока

И так мы рассмотрели следующие вопросы:

- 1. Алгоритмическая структура цикл;
- 2. Виды алгоритмических структур:
 - а. Цикл с предусловием;
 - б. Цикл с постусловием;
 - с. Цикл с параметром;
- 3. Рассмотрели способы записи данных структур;
- 4. Разобрали примеры решения задач с помощью этих структур.

Д.З. №2 стр. 141.

Список использованной литературы

- 1. Информатика и ИКТ. Базовый курс: Учебник для 8 класса. 2-е изд., М.: Бином. Лаборатория знаний, 2007.- 359 с.: ил.
- 2. Преподавание базового курса информатики в средней школе: Методическое пособие. 2-е изд., исп. И доп.- М.: Бином. Лаборатория знаний, 2004.- 540 с.: ил.
- 3. Задачник-практикум. М. Бином. Лаборатория знаний, 2007.- 309 с.: ил.
- 4. Задачник-практикум. Бином. Лаборатория знаний, 2007.- 294 с.: ил.
- 5. <http://pas1.ru/file> Программирование для начинающих.
- 6. <http://www.pascal7.ru/> Все о языке программирования Turbo Pascal.
- 7. <http://tp7.info/> Сайт о языке программирования Turbo Pascal.