*.МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ САМАРСКОЙ ОБЛАСТИ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ПРОФЕССИОЛНАЛЬНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ САМАРСКОЙ ОБЛАСТИ*

*«НОВОКУЙБЫШЕВСКИЙ НЕФТЕХИМИЧЕСКИЙ ТЕХНИКУМ»*

*Российская Федерация, 446202, Самарская область г. Новокуйбышевск, ул. Кирова, 4*

*тел. (84635) 20557 факс 21750*

*e-mail:* [*gapoynnht@yandex.ru*](mailto:gapoynnht@yandex.ru) *сайт:* [*www.nnht.ru*](http://www.nnht.ru/)

### 

### 

## МЕТОДИЧЕСКАЯ РАЗРАБОТКА УРОКА

Дисциплина: Информатика

Тема раздела «Раздел 3. Информационное моделирование»

Тема урока «Понятие алгоритма. Запись линейных алгоритмов на языке программирования (Pascal, Python)»

специальность 18.02.06 Химическая технология органических веществ.

Разработала:

Куразеева С.В.

преподаватель общеобразовательной

дисциплины

г.о. Новокуйбышевск, 2025

### РАССМОТРЕНО

### предметно-цикловой

### комиссией (ПЦК)

### Протокол № \_\_ от «\_\_» \_\_. 2025

### Председатель ПЦК

### Н.П.Комиссарова

### 

## Разработчик:

### ГАПОУ СО «ННХТ» преподаватель С.В. Куразеева

### (место работы) (занимаемая должность) (И.О.Фамилия)

### 

### 

### Рецензенты:

### Зам. дир. по НМР ГАПОУ СО «ННХТ» Щелкова О.Д.

### Председатель ПЦК ГАПОУ СО «ННХТ» Комиссарова Н.П.

#### Тема: Понятие алгоритма. Запись линейных алгоритмов на языке программирования (Pascal, Python)

#### Цель урока:

Формирование у учащихся представления о понятии алгоритма, знакомство с основами языков программирования Pascal и Python, развитие навыков написания линейных алгоритмов.

#### Задачи урока:

**Образовательные:**

1. Дать определение понятиям "алгоритм" и "линейный алгоритм".
2. Привести примеры линейных алгоритмов на языках Pascal и Python.
3. Научить учащихся записывать простые программы на этих языках.

**Развивающие:**

1. Развивать аналитические способности и логическое мышление.
2. Формировать навыки самостоятельного поиска информации и решения проблем.
3. Улучшать навыки работы с компьютером и программированием.

**Воспитательные:**

1. Воспитывать интерес к изучению информатики и программирования.
2. Формировать ответственность за выполнение заданий и работу в команде.

..Развивать культуру общения и взаимодействия в группе.

**Формируемые компетенции:**

ОК 01. Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

ОК 02. Использовать современные средства поиска, анализа и интерпретации

#### Тип занятия:

Комбинированный урок (изложение нового материала + практическая работа).

#### Методы обучения:

* Проблемное обучение
* Словесные методы (рассказ, объяснение)
* Наглядные методы (презентация, демонстрация кода)
* Практические методы (написание кода)

#### Технологии обучения:

* ИКТ (информационно-коммуникационные технологии)
* Практико-ориентированные технологии
* Критическое мышление.

#### Оборудование:

* Компьютеры с установленными IDE для Pascal и Python (например, Lazarus для Pascal и PyCharm для Python).
* Проектор и экран для демонстрации презентации.
* Доступ к интернету для дополнительного поиска информации.

#### Внутрипредметные связи:

* Математика: использование математических операций и функций в алгоритмах
* Физика: моделирование физических процессов с помощью алгоритмов
* Инженерия: применение алгоритмов в проектировании и управлении системами

#### Продолжительность занятия: 90 минут

### План занятия

#### Организационный момент (5 минут)

1. Приветствие учащихся.
2. Проверка готовности к уроку (наличие учебников, тетрадей, компьютеров).
3. Сообщение целей и задач урока.

#### Актуализация знаний учащихся (10 минут)

1. **Вопросы для обсуждения:**

* Что такое алгоритм?
* Какие вы знаете виды алгоритмов?
* Для чего нужны языки программирования?

1. **Мини-викторина:**

* Раздать карточки с вопросами и попросить учащихся дать ответы на них.

#### Сообщение нового материала (30 минут)

1. **Презентация:**

* Понятие алгоритма и его виды (линейный, разветвляющийся, циклический).
* Знакомство с языками Pascal и Python: история, основные особенности, области применения.

1. **Демонстрация кода:**

* Примеры линейных алгоритмов на Pascal и Python.
* Объяснение синтаксиса и структуры программ на данных языках.

1. **Задачи для обсуждения:**

* Какой алгоритм будет линейным? Почему?
* Какие преимущества и недостатки у каждого из языков?

#### Закрепление нового материала (30 минут)

1. **Практическая работа:**

* Написание простых линейных алгоритмов на Pascal и Python.

1. **Групповая работа:**

* Разделение класса на группы по 3-4 человека.
* Каждая группа выбирает одну из задач и выполняет ее на выбранном языке.

1. **Обсуждение результатов:**

* Представление решений от каждой группы.
* Анализ и обсуждение кода.

#### Заключительный этап (15 минут)

1. **Подведение итогов:**

* Обсуждение пройденного материала.
* Ответы на вопросы учащихся.

1. **Домашнее задание:**

* Написать программу на Pascal или Python, которая решает задачу нахождения периметра треугольника по трем сторонам.
* Подготовить краткий доклад о другом виде алгоритмов (например, разветвляющемся или циклическом).

1. **Рефлексия.**

**ХОД ЗАНЯТИЯ**

#### Организационный момент (5 минут)

**Преподаватель:**

* Добрый день, студенты! Сегодня мы будем заниматься необычным способом. Я предложу вам несколько ситуаций, и ваша задача — разобраться, как их решить, используя алгоритмы и программирование. Готовы начать?

**Студенты:**

* Да!

### Актуализация знаний учащихся (10 минут)

**Преподаватель:**

* Представьте себе следующую ситуацию: вы хотите автоматизировать процесс расчета суммы покупки в магазине. Как бы вы это сделали?

**Студент 1:**

* Нужно написать программу, которая будет считать сумму товаров.

**Преподаватель:**

* Верно! Но для начала давайте подумаем, какие шаги нужно выполнить, чтобы получить эту сумму? Какие данные нам нужны?

**Студент 2:**

* Нужно знать цены товаров и количество каждого товара.

**Преподаватель:**

* Отлично! Итак, у нас есть список цен и количество товаров. Какие действия нужно выполнить, чтобы получить общую сумму?

**Студент 3:**

* Нужно перемножить цену каждого товара на его количество и сложить все результаты.

**Преподаватель:**

* Точно! Это и есть алгоритм. Теперь давайте познакомимся с языками программирования, которые позволят нам решать любые алгоритмы с помощью компьютера. Какие языки программирования вы слышали?

**Студенты:**

* Pascal или Python.

Преподаватель:

* Отлично! Теперь давайте проверим ваши знания с помощью мини-викторины. Я раздам вам карточки с вопросами, и вы попробуете ответить на них.

[Раздаются карточки с вопросами.] - см. прил. 1

Преподаватель:

* Хорошо, кто готов поделиться своими ответами?

Обучающийся 3:

* Вопрос: Какой из перечисленных вариантов является примером алгоритма?
  + Ответ: Рецепт приготовления блюда.

Преподаватель:

* Правильно! Это отличный пример алгоритма.

#### Сообщение нового материала (30 минут)

##### Лекционный материал

Преподаватель:

* Теперь давайте поговорим подробнее о понятии алгоритма и его видах. Алгоритм — это последовательность точно определенных действий, приводящих к решению задачи за конечное число шагов.

Алгоритм — это чётко определённая последовательность действий (шагов), которые приводят к решению задачи. Алгоритм должен быть:

* Конечным : состоит из конечного числа шагов.
* Определённым : каждый шаг должен быть чётко понятен.
* Эффективным : алгоритм должен привести к правильному результату за разумное время.

Пример простого алгоритма:

* Задача : Найти сумму двух чисел.
* Алгоритм :
  1. Прочитать первое число a.
  2. Прочитать второе число b.
  3. Вычислить сумму s = a + b.
  4. Вывести результат s.

### 2. Как описывать алгоритмы?

Существует несколько способов записи алгоритмов:

#### a) Натуральный язык

Используется повседневный язык для описания шагов алгоритма. Это самый простой способ, но он не всегда точен.

Пример :  
"Прочитайте два числа, сложите их и выведите результат."

#### b) Блок-схемы

Графический способ описания алгоритмов с использованием стандартных символов:

* Овал — начало/конец.
* Прямоугольник — действия.
* Ромб — проверка условия.
* Стрелки — направление потока управления.

#### c) Псевдокод

Язык, похожий на естественный язык, но ближе к реальным языкам программирования. Псевдокод позволяет описать алгоритм более формально.

Пример псевдокода для суммирования :

plaintext

Copy

начало

прочитать a

прочитать b

s := a + b

вывести s

конец

* Существуют три основных типа алгоритмов:
  + Линейный: последовательное выполнение команд без условий и циклов.
  + Разветвляющийся: содержит условия и различные пути выполнения.
  + Циклический: содержит повторяющиеся действия.

### 3. Способы записи алгоритмов на языках программирования

Теперь давайте рассмотрим, как записать этот же алгоритм на нескольких популярных языках программирования.

Преподаватель:

* Также сегодня мы познакомимся с двумя языками программирования: Pascal и Python.

Преподаватель:

* Pascal:
  + Создан в 1970 году Никлаусом Виртом.
  + Широко используется в образовательных учреждениях для обучения программированию.
  + Преимущества: простота синтаксиса, структурированный подход.

Преподаватель:

* Python:
  + Создан Гвидо ван Россумом в 1991 году.
  + Используется в разработке веб-приложений, научных расчетах, машинном обучении.
  + Преимущества: читаемость кода, богатая библиотека стандартных модулей.

Преподаватель:

* Давайте рассмотрим примеры линейных алгоритмов на обоих языках.

Преподаватель:

* Пример линейного алгоритма на Pascal:

program AreaRectangle;

var

length, width, area: real;

begin

writeln('Введите длину:');

readln(length);

writeln('Введите ширину:');

readln(width);

area := length \* width;

writeln('Площадь прямоугольника: ', area);

end.

Преподаватель:

* Пример линейного алгоритма на Python:

def calculate\_area():

length = float(input("Введите длину: "))

width = float(input("Введите ширину: "))

area = length \* width

print(f"Площадь прямоугольника: {area}")

calculate\_area()

Преподаватель:

* Какой алгоритм будет линейным? Почему?

Обучающийся 4:

* Алгоритм будет линейным, если он выполняется последовательно от начала до конца без условий и циклов.

Преподаватель:

* Совершенно верно!

#### Памятка для студентов: Справочный материал по основному синтаксису Pascal и Python

|  |  |  |
| --- | --- | --- |
| **ЭЛЕМЕНТ ПРОГРАММЫ** | **PASCAL** | **PYTHON** |
| Комментарии | (\* Это комментарий \*) | # Это комментарий |
| Объявление переменной | var x: integer; | x = 0 |
| Ввод данных | readln(x); | x = int(input("Введите значение: ")) |
| Вывод данных | writeln('Значение: ', x); | print("Значение:", x) |
| Оператор присваивания | x := 5; | x = 5 |
| Условный оператор | if x > 0 then writeln('Positive') else writeln('Negative'); | if x > 0: print('Positive') else: print('Negative') |
| Цикл for | for i := 1 to 10 do writeln(i); | for i in range(1, 11): print(i) |
| Цикл while | while x < 10 do begin writeln(x); x := x + 1; end; | while x < 10: print(x); x += 1 |
| Функция | function add(a, b: integer): integer; begin add := a + b; end; | def add(a, b): return a + b |

### Закрепление нового материала (30 минут)

Преподаватель:

* Теперь давайте закрепим материал на практике. Разделитесь на группы и попробуйте решить следующие задачи.
* Студенты разбиваются на группы и начинают работать над заданиями.

Задача 1: Расчет концентрации раствора

Описание: В лаборатории необходимо приготовить раствор определенной концентрации. Напишите программу, которая принимает массу растворяемого вещества и объем раствора и вычисляет его концентрацию в граммах на литр (г/л).

Python

def calculate\_concentration(mass, volume):

concentration = mass / volume

return concentration

mass = float(input("Введите массу растворяемого вещества (г): "))

volume = float(input("Введите объем раствора (л): "))

concentration = calculate\_concentration(mass, volume)

print(f"Концентрация раствора составляет {concentration:.2f} г/л")

Pascal

program CalculateConcentration;

var

mass, volume, concentration: real;

begin

writeln('Введите массу растворяемого вещества (г): ');

readln(mass);

writeln('Введите объем раствора (л): ');

readln(volume);

concentration := mass / volume;

writeln('Концентрация раствора составляет ', concentration:0:2, ' г/л');

end.

Задача 2: Конвертация единиц измерения температуры

Описание: При проведении экспериментов часто требуется переводить температуру из градусов Цельсия в градусы Фаренгейта и обратно. Напишите программу, которая выполняет эти преобразования.

Python:

def celsius\_to\_fahrenheit(celsius):

return celsius \* 9/5 + 32

def fahrenheit\_to\_celsius(fahrenheit):

return (fahrenheit - 32) \* 5/9

temperature = float(input("Введите температуру: "))

scale = input("Выберите шкалу ('C' для Цельсия или 'F' для Фаренгейта): ")

if scale == 'C':

print(f"Температура в Фаренгейтах: {celsius\_to\_fahrenheit(temperature):.2f}")

elif scale == 'F':

print(f"Температура в Цельсиях: {fahrenheit\_to\_celsius(temperature):.2f}")

else:

print("Неверная шкала.")

Pascal:

program TemperatureConversion;

var

temperature: real;

scale: char;

celsius, fahrenheit: real;

begin

writeln('Введите температуру: ');

readln(temperature);

writeln('Выберите шкалу ("C" для Цельсия или "F" для Фаренгейта): ');

readln(scale);

if scale = 'C' then

begin

fahrenheit := temperature \* 9/5 + 32;

writeln('Температура в Фаренгейтах: ', fahrenheit:0:2);

end

else if scale = 'F' then

begin

celsius := (temperature - 32) \* 5/9;

writeln('Температура в Цельсиях: ', celsius:0:2);

end

else

begin

writeln('Неверная шкала.');

end;

end.

Задача 3: Вычисление плотности раствора

Описание: Плотность раствора может быть важным параметром в химических экспериментах. Напишите программу, которая принимает массу раствора и его объем и вычисляет плотность раствора в г/см³.

Python:

def calculate\_density(mass, volume):

density = mass / volume

return density

mass = float(input("Введите массу раствора (г): "))

volume = float(input("Введите объем раствора (см³): "))

density = calculate\_density(mass, volume)

print(f"Плотность раствора составляет {density:.2f} г/см³")

Pascal:

program CalculateDensity;

var

mass, volume, density: real;

begin

writeln('Введите массу раствора (г): ');

readln(mass);

writeln('Введите объем раствора (см³): ');

readln(volume);

density := mass / volume;

writeln('Плотность раствора составляет ', density:0:2, ' г/см³');

end.

profile

Qwen2.5-Plus 10:32 pm

Конечно, вот еще несколько простых задач, которые также связаны с химической технологией органических веществ и могут быть решены с помощью программирования на языках Python и Pascal.

Задача 4: Вычисление плотности раствора

Описание: Плотность раствора может быть важным параметром в химических экспериментах. Напишите программу, которая принимает массу раствора и его объем и вычисляет плотность раствора в г/см³.

Python:

python

Copy

1

2

3

4

5

6

7

8

9

⌄

def calculate\_density(mass, volume):

density = mass / volume

return density

mass = float(input("Введите массу раствора (г): "))

volume = float(input("Введите объем раствора (см³): "))

density = calculate\_density(mass, volume)

print(f"Плотность раствора составляет {density:.2f} г/см³")

Pascal:

pascal

Copy

1

2

3

4

5

6

7

8

9

10

11

12

13

14

program CalculateDensity;

var

mass, volume, density: real;

begin

writeln('Введите массу раствора (г): ');

readln(mass);

writeln('Введите объем раствора (см³): ');

readln(volume);

density := mass / volume;

writeln('Плотность раствора составляет ', density:0:2, ' г/см³');

end.

Задача 4: Расчет количества вещества по молярной концентрации

Описание: В лаборатории часто требуется рассчитать количество вещества (в молях) по известной молярной концентрации и объему раствора. Напишите программу, которая выполняет этот расчет.

Python:

def calculate\_moles(concentration, volume):

moles = concentration \* volume

return moles

concentration = float(input("Введите молярную концентрацию раствора (М): "))

volume = float(input("Введите объем раствора (л): "))

moles = calculate\_moles(concentration, volume)

print(f"Количество вещества составляет {moles:.2f} моль")

Pascal:

program CalculateMoles;

var

concentration, volume, moles: real;

begin

writeln('Введите молярную концентрацию раствора (М): ');

readln(concentration);

writeln('Введите объем раствора (л): ');

readln(volume);

moles := concentration \* volume;

writeln('Количество вещества составляет ', moles:0:2, ' моль');

end.

Преподаватель:

* Все группы отлично справились со своими задачами. Давайте обсудим возможные ошибки и как их избежать.

#### Заключительный этап (15 минут)

Преподаватель:

* Давайте подведем итоги урока. Что нового вы узнали сегодня?

Обучающийся 5:

* Мы узнали, что такое алгоритм и как записывать простые программы на Pascal и Python.

Преподаватель:

* Верно. Есть ли у вас вопросы?

Студент 6:

* Можно ли использовать линейные алгоритмы для решения более сложных задач?

Преподаватель:

* Конечно, но для более сложных задач часто используются другие типы алгоритмических структур, такие как ветвления и циклы. Ваше домашнее задание будет включать решение задач с использованием линейных алгоритмов. Напишите программы на Pascal и Python для следующих задач:
  + Вычисление объема параллелепипеда (даны длина, ширина и высота).
  + Конвертация градусов Фаренгейта в градусы Цельсия.
  + Вычисление площади трапеции (даны основания и высота).
  + Вычисление времени, необходимого для преодоления определенного расстояния при известной скорости.
  + Вычисление стоимости покупки с учетом налога (даны цена товара и процент налога).

Преподаватель:

* Также рекомендую вам изучить дополнительные ресурсы для углубленного изучения темы. Например, решать задачи на онлайн-платформах Codeforces или LeetCode.

Студенты:

* Спасибо за урок!

Преподаватель:

* На этом наш урок окончен. До встречи на следующем занятии!

ПРИЛОЖЕНИЕ 1

### Карточки для мини-викторины

#### Карточка 1

Вопрос: Какой из перечисленных вариантов является примером алгоритма?

* a) Рецепт приготовления блюда
* b) Прогулка по парку
* c) Чтение книги

Правильный ответ: a) Рецепт приготовления блюда

#### Карточка 2

Вопрос: Назовите три основных типа алгоритмов.

* a) Линейный, разветвляющийся, циклический
* b) Простой, сложный, смешанный
* c) Математический, физический, химический

Правильный ответ: a) Линейный, разветвляющийся, циклический

#### Карточка 3

Вопрос: Какие языки программирования вы уже знаете?

* a) C++, Java, Python
* b) HTML, CSS, JavaScript
* c) Pascal, Basic, Fortran

Правильный ответ: a) C++, Java, Python (или любые другие варианты, которые учащиеся действительно знают)

#### Карточка 4

Вопрос: Что делает оператор writeln в Pascal?

* a) Выводит текст на экран
* b) Считывает данные с клавиатуры
* c) Объявляет переменную

Правильный ответ: a) Выводит текст на экран

#### Карточка 5

Вопрос: Что делает оператор print в Python?

* a) Выводит текст на экран
* b) Считывает данные с клавиатуры
* c) Объявляет переменную

Правильный ответ: a) Выводит текст на экран

#### Карточка 6

Вопрос: Как правильно объявить переменную в Pascal?

* a) var x: integer;
* b) x = 0;
* c) int x;

Правильный ответ: a) var x: integer;

#### Карточка 7

Вопрос: Как правильно объявить переменную в Python?

* a) var x: integer;
* b) x = 0;
* c) int x;

Правильный ответ: b) x = 0;

#### Карточка 8

Вопрос: Какая команда используется для ввода данных с клавиатуры в Pascal?

* a) readln(x);
* b) input("Введите значение: ")
* c) scanf("%d", &x);

Правильный ответ: a) readln(x);

#### Карточка 9

Вопрос: Какая команда используется для ввода данных с клавиатуры в Python?

* a) readln(x);
* b) input("Введите значение: ")
* c) scanf("%d", &x);

Правильный ответ: b) input("Введите значение: ")

#### Карточка 10

Вопрос: Какой оператор используется для условного выполнения кода в Pascal?

* a) if x > 0 then writeln('Positive') else writeln('Negative');
* b) if x > 0: print('Positive') else: print('Negative')
* c) switch(x) { case 1: printf("Positive"); break; }

Правильный ответ: a) if x > 0 then writeln('Positive') else writeln('Negative');

#### Карточка 11

Вопрос: Какой оператор используется для условного выполнения кода в Python?

* a) if x > 0 then writeln('Positive') else writeln('Negative');
* b) if x > 0: print('Positive') else: print('Negative')
* c) switch(x) { case 1: printf("Positive"); break; }

Правильный ответ: b) if x > 0: print('Positive') else: print('Negative')

#### Карточка 12

Вопрос: Какой цикл используется для выполнения определенного количества итераций в Pascal?

* a) for i := 1 to 10 do writeln(i);
* b) for i in range(1, 11): print(i)
* c) while i < 10: print(i); i += 1

Правильный ответ: a) for i := 1 to 10 do writeln(i);

#### Карточка 13

Вопрос: Какой цикл используется для выполнения определенного количества итераций в Python?

* a) for i := 1 to 10 do writeln(i);
* b) for i in range(1, 11): print(i)
* c) while i < 10: print(i); i += 1

Правильный ответ: b) for i in range(1, 11): print(i)

#### Карточка 14

Вопрос: Какой цикл используется для выполнения действий до тех пор, пока условие истинно в Pascal?

* a) while x < 10 do begin writeln(x); x := x + 1; end;
* b) while x < 10: print(x); x += 1
* c) do { printf("%d\n", x); x++; } while (x < 10);

Правильный ответ: a) while x < 10 do begin writeln(x); x := x + 1; end;

#### Карточка 15

Вопрос: Какой цикл используется для выполнения действий до тех пор, пока условие истинно в Python?

* a) while x < 10 do begin writeln(x); x := x + 1; end;
* b) while x < 10: print(x); x += 1
* c) do { printf("%d\n", x); x++; } while (x < 10);

Правильный ответ: b) while x < 10: print(x); x += 1

ПРИЛОЖЕНИЕ 2

#### 

|  |  |  |
| --- | --- | --- |
| **ЭЛЕМЕНТ ПРОГРАММЫ** | **PASCAL** | **PYTHON** |
| Комментарии | (\* Это комментарий \*) | # Это комментарий |
| Объявление переменной | var x: integer; | x = 0 |
| Ввод данных | readln(x); | x = int(input("Введите значение: ")) |
| Вывод данных | writeln('Значение: ', x); | print("Значение:", x) |
| Оператор присваивания | x := 5; | x = 5 |
| Условный оператор | if x > 0 then writeln('Positive') else writeln('Negative'); | if x > 0: print('Positive') else: print('Negative') |
| Цикл for | for i := 1 to 10 do writeln(i); | for i in range(1, 11): print(i) |
| Цикл while | while x < 10 do begin writeln(x); x := x + 1; end; | while x < 10: print(x); x += 1 |
| Функция | function add(a, b: integer): integer; begin add := a + b; end; | def add(a, b): return a + b |

**ПРИЛОЖЕНИЕ 3**

        