



Программирование, лекция 1



Кафедра ИУ7 МГТУ им. Н. Э. Баумана,
2021 год



История МГТУ им. Н. Э. Баумана

1830 - создание Московского ремесленного учебного заведения (МРУЗ)

1868 - преобразование в Императорское московское техническое училище (ИМТУ)

1917 - Московское высшее техническое училище (МВТУ)

1930-е, 40-е годы - реформы, присвоение имени Баумана

1955 - создание факультета "Приборостроение" (П)

1989 - преобразование в Московский государственный технический университет (МГТУ им. Н. Э. Баумана), создание кафедры ИУ-7

Учебный план

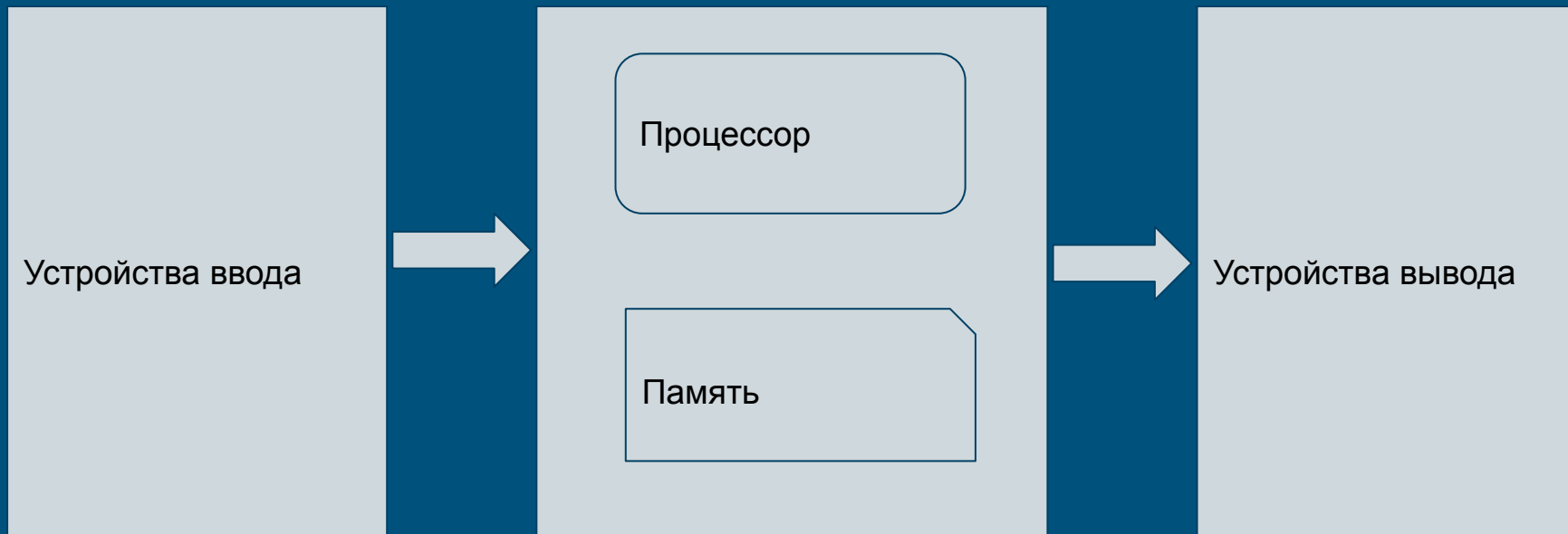
№	Название дисциплины		Отрезки по годам:					
			1	2	3	4	5	6
Б1	Дисциплины (модули)		2021/22		2022/23		2023/24	
Обязательная часть								
1	Аналитическая геометрия 2021	ФН12	4					
2	Инженерная графика 2021	ИУ7	5					
3	Иностранный язык 2021	Л2	2	2	2	2	2	2
4	Математический анализ 2021	ФН12	5					
5	Программирование 2021	ИУ7	10	6	6			
6	Физическая культура и спорт 2021	ФВ	2					
7	Интегралы и дифференциальные уравнения 2021	ФН12		5				
8	История 2021	СГН1		3				
9	Линейная алгебра и функция нескольких переменных 2021	ФН12		4				
10	Физика 2021	ФН4		4	5			
11	Дискретная математика 2021	ФН12			4			
12	Типы и структуры данных 2021	ИУ7			3			
13	Правоведение 2021	ЮР3			3			
14	Логика и теория алгоритмов 2021	ФН12				3		
15	Архитектура ЭВМ 2021	ИУ6			3		4	
16	Политология 2021	СГН3				3		
17	Социология 2021	СГН2				3		
18	Теория вероятностей 2021	ФН12					4	
19	Операционные системы 2021	ИУ7					5	
20	Математическая статистика 2021	ФН12						
21	Основы возможных отношений 2021	СГН2						
22	Экология 2021	Э9						
23	Безопасность жизнедеятельности 2021	Э9						
24	Философия 2021	СГН4						
25	Экономика часть 1 2021	ИБМ3						
26	Экономика часть 2 2021	ИБМ6						
Часть, формируемая участниками образовательных отношений								
1	Теоретическая информатика 2021	ИУ7	2					
2	Основы программной инженерии 2021	ИУ7		4				
3	Основы электроники 2021	ИУ7			3			
4	Машинно-зависимые языки программирования 2021	ИУ7				4		
5	Объектно-ориентированное программирование 2021	ИУ7				4		
6	Компьютерная графика 2021	ИУ7				4	2	
7	Вычислительные алгоритмы 2021	ИУ7				3		
8	Базы данных 2021	ИУ7					6	
9	Анализ алгоритмов 2021	ИУ7					3	
10	Моделирование 2021	ИУ7						
11	Проектирование программного обеспечения 2021	ИУ7						
12	Функциональное и логическое программирование 2021	ИУ7						

Компьютер. ЭВМ

Компьютер - устройство, способное выполнять заданную, чётко определённую, изменяемую последовательность операций (численные расчёты, преобразование данных и т. д.)

Электронно-вычислительная машина - основной вид реализации компьютеров, который технически выполнен на электронных элементах

Схема ЭВМ



Элементарные термины

Процессор — интегральная схема, исполняющая машинные инструкции (код программ), главная часть аппаратного обеспечения компьютера

Машинный код — система команд (набор кодов операций) конкретной вычислительной машины, которая интерпретируется непосредственно процессором. Кодирована в двоичном виде

Файл — поименованное место на диске

Алгоритм — конечная совокупность точно заданных правил решения некоторого класса задач или набор инструкций, описывающих порядок действий исполнителя для решения определённой задачи.

Программа

Исполняемая программа — комбинация компьютерных инструкций и данных, позволяющая аппаратному обеспечению вычислительной системы выполнять вычисления или функции управления

Исходный текст программы — синтаксическая единица, которая соответствует правилам определённого языка программирования, состоящая из определений и операторов или инструкций, необходимых для определённой функции, задачи или решения проблемы

Языки программирования

Язык программирования — формальный язык, предназначенный для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, определяющих действия, которые выполнит ЭВМ под её управлением.

Способы классификации:

- по уровню абстракции от аппаратной части
- по способу выполнения готовой программы
- по используемой парадигме программирования

Классификация языков программирования

- По уровню абстракции от аппаратной части:
 - низкоуровневые
 - высокоуровневые
- По способу выполнения исполняемой программы:
 - компилируемые
 - интерпретируемые
- По парадигме программирования:
 - императивные / процедурные языки
 - аппликативные / функциональные языки
 - языки системы правил / декларативные языки
 - объектно-ориентированные языки

Исполняемый файл

Исполняемый файл - файл, содержащий программу в виде, в котором она может быть исполнена компьютером (то есть в машинном коде).

Получение исполняемых файлов требует выполнения компиляции.

Компилятор - программа для преобразования исходного текста другой программы на определённом языке в объектный модуль (файл с машинным кодом).

Язык программирования Python

Python - высокоуровневый язык программирования общего назначения. Интерпретируемый. Является полностью объектно-ориентированным.

Программа

модули

операторы

выражения

объекты

Лексемы языка Python

Символы алфавита любого языка программирования образуют **лексемы**. По умолчанию - кодировка UTF-8.

Лексема (token) – это минимальная единица языка, имеющая самостоятельный смысл. Лексемы формируют базовый словарь языка, понятный компилятору.

Всего существует пять видов лексем:

- ключевые слова (keywords)
- идентификаторы (identifiers)
- литералы (literals)
- операции (operators)
- знаки пунктуации (разделители, punctuators)

Строки программы

Физическая строка исходного файла – это строка заканчивающаяся символом признака конца строки.

Программа Python разделена на несколько логических строк. Логическая строка содержит одну или более физических строк, соединяющихся правилами языка.

Ведущие пробельные символы (пробелы и табуляции) в начале строки используются в Python для определения группы инструкций, как единого целого – составной инструкции или блока.

Комментарии в Python начинаются с символа "#".

Объединение физических строк: явное (через "\") и неявное.

Ключевые слова

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield