## Университет ИТМО

## ИТИП

## История науки и техники

## Реферат

## Тема: “ История возникновения программирования”

Студент: Величко Максим Иванович

Группа: М32061

Поток: ИНТ 6.2

Санкт-Петербург 2022

**Оглавление**

[Введение 3](#_bookmark0)

[История возникновения программирования. Первые языки](#_bookmark1) [программирования. 3](#_bookmark1)

[Структурное и модульное программирование. 5](#_bookmark2)

[Объектно-ориентированное программирование 5](#_bookmark3)

[Компонентное программирование 6](#_bookmark4)

[Заключение 7](#_bookmark5)

[Литература 7](#_bookmark6)

# Введение

Компьютерные технологии с каждым днем все глубже и глубже входят в нашу жизнь и уже являются ее неотъемлемой частью. Процесс проникновения компьютеров в нашу жизнь определил разработку новых знаковых систем для записи алгоритмов – языков программирования. Смысл их появления – упрощения программного кода. Для облегчения взаимодействия с компьютерами создается все новое ПО с помощью различных языков программирования.

Целью, данной данного реферата, является изучение истории возникновения программирования и основных принципов и подходов при создании языка программирования.

# История возникновения программирования. Первые языки программирования.

С древности известны попытки создать приспособления, ускоряющие и облегчающие процесс вычисления. В Древней Греции и Риме применяли устройства – абак похожее на счеты. Подобные устройства были широко распространены и в странах Древнего Востока. немецкие ученые В. Шиккард, Г.Лейбниц и французский ученый Б. Паскаль в XV в. создали прототипы устройства - арифмометр. Вычислительная техника улучшалась в течение многих веков, но все еще понятия «программа и программирование» не применялось.

В 1830 году анализирую обработку переписи населения во Франции, английский ученый Чарльз Бэббидж, теоретически исследовал процесс выполнения и обосновал основы архитектуры вычислительной машины. Работая над аналитической машиной, он

предопределил многие принципы работы современных ЭВМ, в частности принцип программного управления и запоминаемой программы. В 1843 ученой Аде Лавлейс была разработана первая программа для аналитической машины Бэббиджа и уверила его в использовании бинарной системы счисления вместо десятичной, разработала первые принципы программирования, которые предусматривали повторения кусков команды при определенных условиях. Именно ею были предложены термины «рабочая ячейка» и «цикл».

В 1854г Джордж Буль опубликовал книгу «Законы мышления», в которой развивал теорию алгебры логики – Булеву Алгебру. На основе булевой алгебры в конце XIXв. построена теория конструирование дискретных автоматов и релейно-контактных схем. Булева алгебра оказала огромное влияние на развитие ЭВМ, будучи инструментом анализа и разработки сложных схем.

В 1936г А. Тьюрингом было введено понятие машина Тьюринга, как уточнение интуитивного понятия алгоритма. Ученый доказывал, что любой алгоритм, может быть, в ней выполнен, то есть доказал возможность создания универсальной ЭВМ.

Логические схемы вычислительных машин были разработаны в конце 1940-х гг. Дж. фон Нейманом, Г. Гольдстайном и А. В. Берксом. Особый вклад в эту работу внес американский математик Джон фон Нейман, принимавший участие в создании ЭНИАК. Им была

предложена идея хранения команд управления и данных в машинной памяти и им же были сформулированы основные принципы построения современных ЭВМ. ЭВМ с хранимой программой оказались более быстродействующими и гибкими, чем ранее созданные.

В 1951 г. в США было налажено первое серийное производство электронных машин УНИВАК (универсальная автоматическая вычислительная машина). В это же время фирма IBM начала серийный выпуск машины IBM/701.

В СССР первыми авторами ЭВМ, изобретенной в 1948 г., являются И. С. Брук и Б. И. Рамеев. А первая советская ЭВМ с сохраняющейся программой создана в 1951 г. под руководством С. А Лебедева (МЭСМ -- малая электронная счетная машина). В 1953 г. в Советском Союзе начался серийный выпуск машин, первыми их которых были БЭСМ-1,

«Стрела».

С появлением цифровых программно-управляемых машин родилась новая область прикладной математики – программирование. Как область науки и профессия она возникла в 1950-х гг. Изначально программы писались вручную на машинном коде. Программы были очень большими, а их отладка – очень трудоемкой. Для упрощения написания программ были разработаны мнемокоды, по структуре схожие с машинным кодом, использующие символьную адресацию.

До конца 1950-х гг. ЭВМ основным элементом конструкции были электронные лампы. В 1954 г. был создан Fortran (Formula Translation) -- первый язык программирования высокого уровня, созданный под руководством Дж. Бэкуса, используемый до настоящего времени в различных модификациях. Так, в 1965 г. была разработана упрощенная версия Фортрана -- Basic. В 1966 г. комиссия Американской ассоциации стандартов (ASA) разработала два стандарта языка: Фортран и Базисный Фортран.

В 1953 г. А.А.Ляпунов предложил операторный метод программирования, который заключался в автоматизации программирования, по средствам решения задачи в виде совокупности операторов, образующих логическую схему задачи. Схемы позволяли расчленить громоздкий процесс составления программы, части которой составлялись по формальным правилам, а затем объединялись в целое. Для проверки идей операторного метода в СССР в 1954 г. была разработана первая программирующая программа ПП-1, а в 1955 г. более совершенная - ПП-2.

В США в 1954 г. стали применять алгебраический подход, подобный, по существу, с операторным методом. В 1956 г. корпорация IBM разработала универсальную ПП Фортран для автоматического программирования на ЭВМ IBM/704.

В этот период по мере накопления опыта и теоретического осмысления совершенствовались языки программирования. В 1958-1960 гг. в Европе был создан ALGOL, который породил целую серию алголоподобных языков: Algol W, (1967), Algol 68, Pascal (Н. Вирт, 1970 г.), С (Д. Ритчи и Б. Керниган, 1972 г.), Ada (под руководством Ж. Ишбиа, 1979 г.), C++ (1983).

На начало 1970-х гг. существовало более 700 языков высокого уровня и около 300 трансляторов для автоматизации программирования.

# Структурное и модульное программирование.

В 1965 году итальянцами Бомом и Джакопини было предложено в качестве базовых алгоритмических элементов следование, ветвление, и цикл. 1970-х годах данная методология была оформлена, и компания IBM сообщила об использовании «Усовершенствованных методов программирования» и сформулиоровала следующие принципы:

* сложная задача разбивается на простые, функционально управляемые задачи, каждая задача имеет один вход и один выход; управляющий поток программы состоит из совокупности элементарных функциональных подзадач;
* управляющие структуры просты, т. е. логическая задача должна состоять из минимальной, функционально полной совокупности достаточно простых управляющих структур;
* программа разрабатывается поэтапно, на каждом этапе решается ограниченное число точно поставленных задач.

Четко сформулированные основы нисходящей разработки, структурного кодирования и сквозного контроля позволяли перейти к промышленным методам разработки программного обеспечения.

# Объектно-ориентированное программирование

На смену структурному программированию в начале 1990-г пришло объектно-ориентированное программирование – ООП. Его можно рассматривать как модульное программирование нового уровня, когда вместо во многом случайного, механического объединения процедур и данных главным становится их смысловая связь. Объект рассматривается как логическая единица, которая содержит данные и правила (методы) их обработки.

ООП основано на трех важнейших принципах (инкапсуляция, наследование, полиморфизм), придающих объектам новые свойства.

1. Инкапсуляция -- объединение в единое целое данных и алгоритмов их обработки. Данные здесь -- поля объекта, а алгоритмы -- объектные методы.
2. Наследование -- свойство объектов порождать своих потомков. Объект-потомок автоматически наследует все поля и методы, может дополнять объекты новыми полями, заменять и дополнять методы.
3. Полиморфизм -- свойство родственных объектов решать схожие по смыслу проблемы разными способами.

На современном этапе развиваются инструментальные среды и системы визуального программирования для создания программ на языках высокого уровня: (Turbo Pascal, Delphi, Visual Basic, C++ и др.).

# Компонентное программирование

Развитие основных принципов объектно-ориентированного программирования получило с появлением компонентного программирования (КП). КП - динамический процесс без

жестких правил, выполняющийся в основном для распределенной разработки (программирования) распределенных систем. Суть КП в том, что независимые проектировщики, программисты разрабатывают независимые компоненты (отдельные части) единой системы, распределенные по множеству узлов большой сети. Эти части могут принадлежать разным собственникам и управляться организационно независимыми

администраторами.

В КП компонент рассматривается как хранилище (в виде DLL-или ЕХЕ файлов) для одного или нескольких классов. Классы распространяются в бинарном виде, а не в виде исходного кода. Предоставление доступа к методам класса осуществляется через строго определенные интерфейсы по протоколу. Это снимает проблему несовместимости компиляторов, обеспечивая без перекомпиляции смену версий классов в разных приложениях. Интерфейсы задают содержание сервиса и являются посредником между клиентом и сервером.

# Заключение

В процессе написания данного реферата мною была изучена история возникновения самого программирования. Также была проведена систематизация знаний о подходах и принципах создании новых языков программирования. В заключении следует отметить, что рассмотренная тема, позволяет просмотреть путь становления технологий и языков программирования и является интересной с точки зрения специалиста в области

информационных технологий.

# Литература

* 1. Шмокин, М. Н. История развития принципов счета, вычислительной техники и программирования: учебное пособие / М. Н. Шмокин. — Пенза: ПензГТУ, 2014. — 159 с
  2. Барков, И. А. Объектно-ориентированное программирование: учебник / И. А. Барков. — Санкт-Петербург: Лань, 2019. — 700 с. — ISBN 978-5-8114-3586-9.
  3. Информатика. Информационные основы средств вычислительной техники: учебное пособие / Е. В. Грачёва, О. С. Литвинская, Н. Н. Короткова, И. А. Казакова ; под редакцией А. П. Ремонтова. — Пенза: ПензГТУ, 2011. — 166 с.
  4. Догадин, Н. Б. Архитектура компьютера: учебное пособие / Н. Б. Догадин. — 4-е изд. — Москва : Лаборатория знаний, 2020. — 274 с. — ISBN 978-5-00101-662-5.