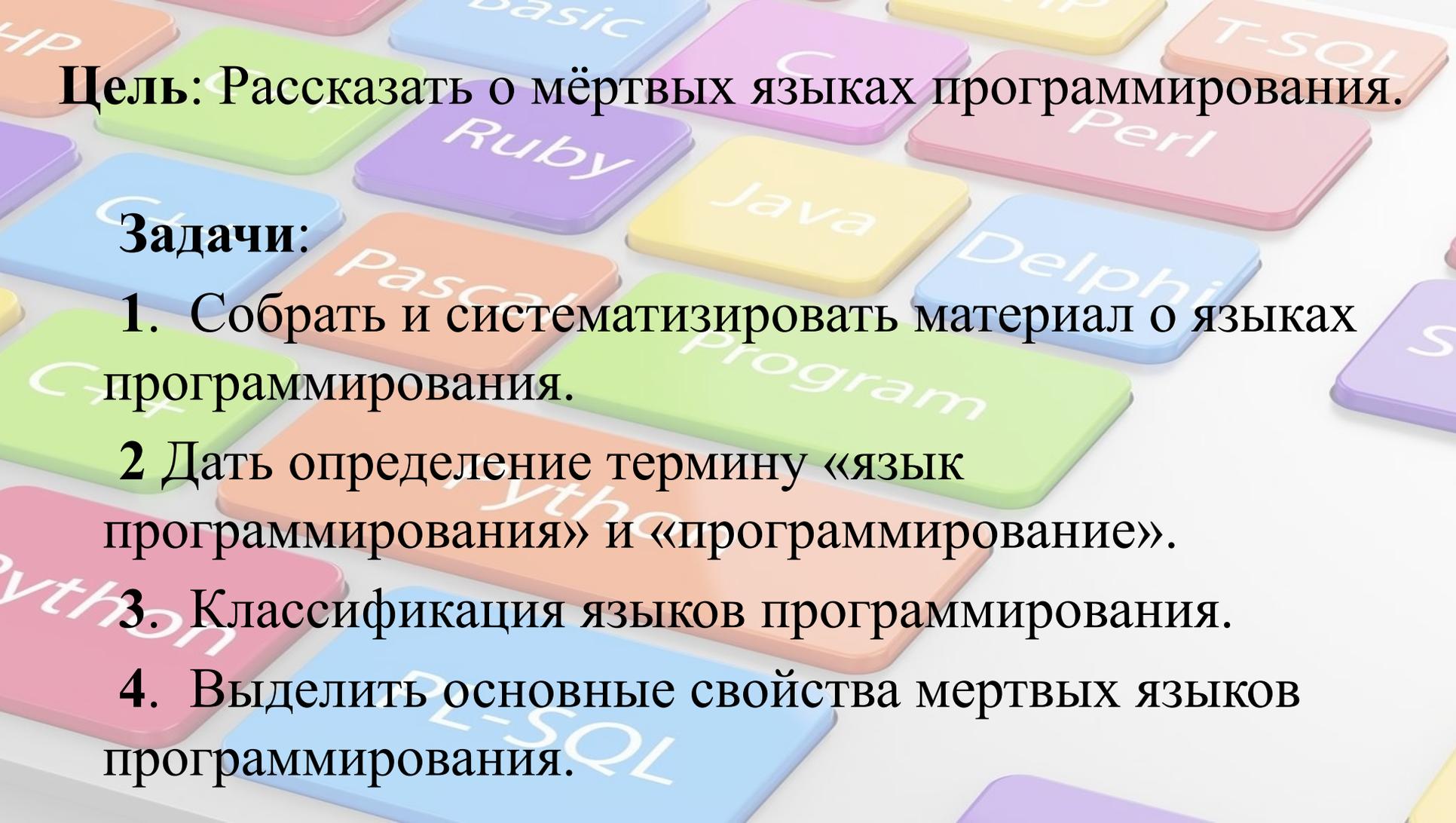


«Мертвые языки программирования»

**Выполнил:
Ученик 11г класса
Тытянчук Даниил**



Цель: Рассказать о мёртвых языках программирования.

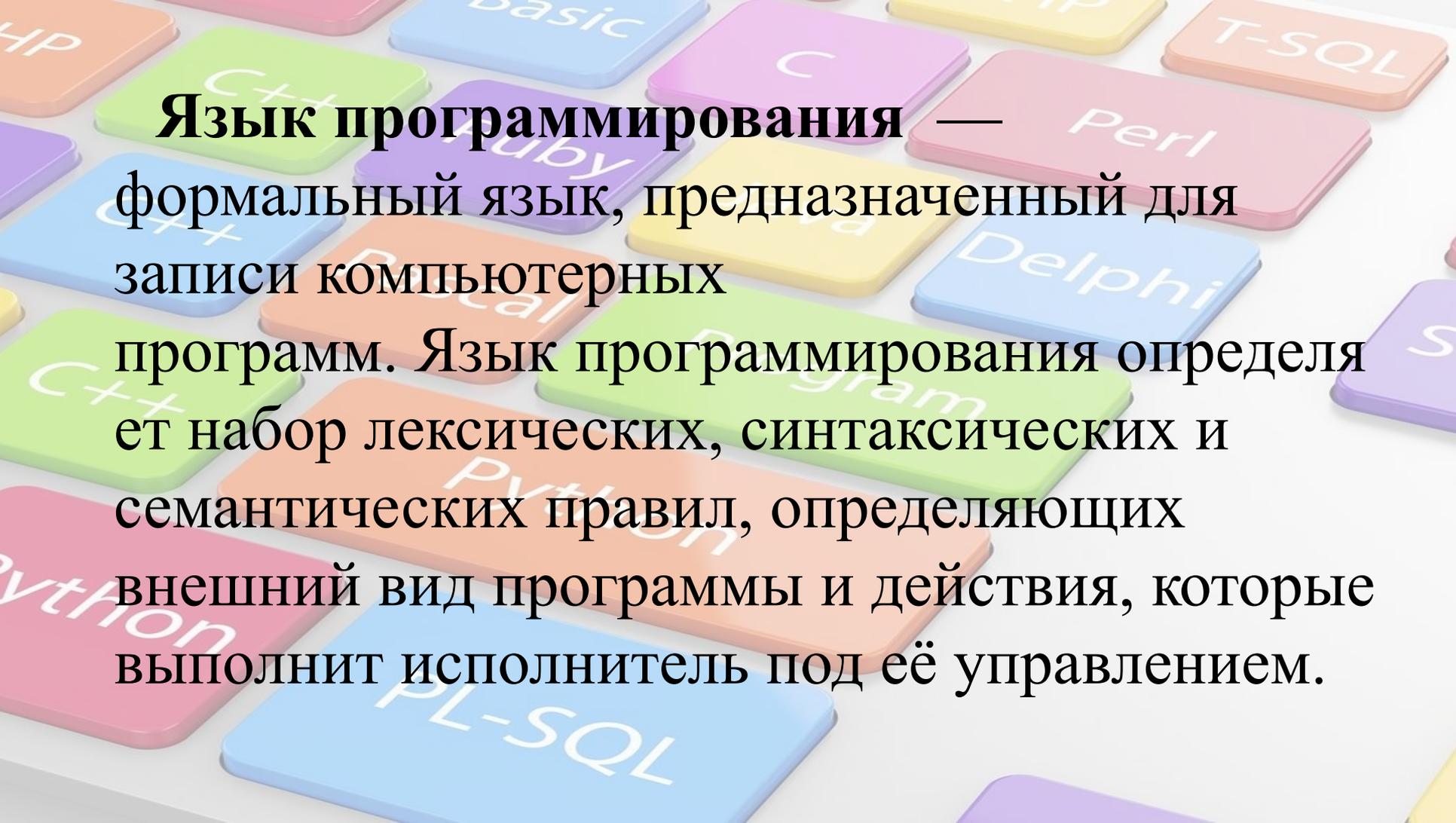
Задачи:

1. Собрать и систематизировать материал о языках программирования.
2. Дать определение термину «язык программирования» и «программирование».
3. Классификация языков программирования.
4. Выделить основные свойства мертвых языков программирования.

Актуальность проекта:

К выбору данной теме меня побудило незнакомое для меня понятие «Мертвые языки программирования». Данная проектная работа позволяет приблизиться к истокам программирования и дать ответ на основной вопрос: почему одни языки программирования так и не проявили себя и про них все забыли, а другие легли в основу новых, которые востребованы и позволяют людям развивать информационные технологии?

Объект исследования: Языки программирования.



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

Поколения ЭВМ	Характеристика ЭВМ	Языки программирования	Характеристика ЯП.
1 поколение (1946-1954г.г.)	Для научно-технических расчетов; носитель информации - перфокарты и перфоленты; программное обеспечение-библиотека стандартных программ, автокоды.	Машинные	Ориентированы на использование в конкретной ЭВМ, сложны в освоении, требуют хорошего знания архитектуры ЭВМ.
2 поколение (1958-1960г.г.)	Для технологических и экономических расчётов; носитель информации - магнитная лента; созданы трансляторы.	Ассемблеры, макроассемблеры	Более удобны для использования, но по-прежнему машинно-зависимы
3 поколение (1960-1966г.г.)	Для управления и технологических расчётов; носитель информации - диск; программное обеспечение- операционные системы; СУБД; пакеты прикладных программ.	Языки высокого уровня	Мобильные, человеко - ориентированные, проще в освоении
4 поколение (1970-1980г.г.)	Для информационного обслуживания, телекоммуникации; носитель информации - гибкий диск; прикладное ПО общего назначения; сетевое ПО; мультимедия.	Непроцедурные, объектно-ориентированные, языки запросов, параллельные	Ориентированы на непрофессионального пользователя и на ЭВМ с параллельной архитектурой
5 поколение (1980г.- по наше время)	Телекоммуникации, информационное обслуживание; носитель информации - гибкий, жёсткий и оптический диски; интеллектуальные программные системы;	Телекоммуникации, информационное обслуживание; носитель информации - гибкий, жёсткий и оптический диски; интеллектуальные программные системы;	Ориентированы на повышение интеллектуального уровня ЭВМ и интерфейса с языками



Разработчик: Джон Кемени, 1964.

Basic — первый демократичный язык программирования. Он был создан как упрощенный аналог Fortran и предназначался для людей, которые не имели отношения к науке, но хотели научиться программировать.

Язык стал популярным в эпоху микрокомпьютеров — у первых устройств было слишком мало памяти для компиляции «настоящих» языков программирования. В то же время урезанному компилятору Basic требовалось всего 2 КБ. Basic стал лингва франка для начинающих программистов: если в 1970-х вы программировали дома, то, вероятно, писали именно на этом языке.

Pascal

Язык программирования

Разработчик: Никлаус Вирт, 1970

Язык был создан на основе Algol-60, идеи которого, по мнению Вирта, не нашли отражения в Algol-68. Сначала Pascal стал популярен как «язык введения в Computer Science», но к 1980 году стал вторым по популярности в компьютерной сети Usenet. Вирт рассматривал все семейство языков — Pascal, Modula и Oberon — как единый языковой концепт.



COBOL PROGRAMMING

Cobol возник благодаря тому, что научные и бизнес подразделения IBM использовали разные языки программирования. 70 лет назад высокоуровневые языки предназначались либо для инженерных вычислений, либо для управления данными. Если в научной среде стандартом был Fortran, среди бизнеса единства не было — компании использовали Comtran, Flow-Matic и другие.

В 1960 году Министерство обороны США организовало комитет по разработке единого универсального языка программирования для бизнес - задач — им стал Cobol.

Cobol был одним из четырех «материнских» языков, наряду с Algol, Fortran и LISP. Сегодня он почти забыт, но когда-то был самым популярным языком в мире и на нем по-прежнему работают многие устаревшие бизнес - системы.



Разработчик: Комитет Algol, 1960.

Из четырех материнских языков Algol — самый «мертвый». LISP и Cobol до сих пор на слуху, поскольку на них работает множество устаревших систем, а Fortran иногда используется в научных целях. Но я встречал множество программистов, которые, ни разу не слышали об Algol — при этом по степени влияния на современные языки с ним может сравниться разве что LISP.

«APL»

Разработчик: Кен Айверсон, 1962.

В оригинале APL — написанная от руки нотация для математических массивов, которую IBM взяла за основу для создания языка программирования. Язык использовался для обработки массивов — и позволял сравнительно короткими командами манипулировать большими блоками чисел.

Если вы раньше слышали об APL, то, скорее всего, знаете его как «этот странный язык символов».

APL использует собственные символы, поэтому для него нужна специальная клавиатура.

«Simula 67»

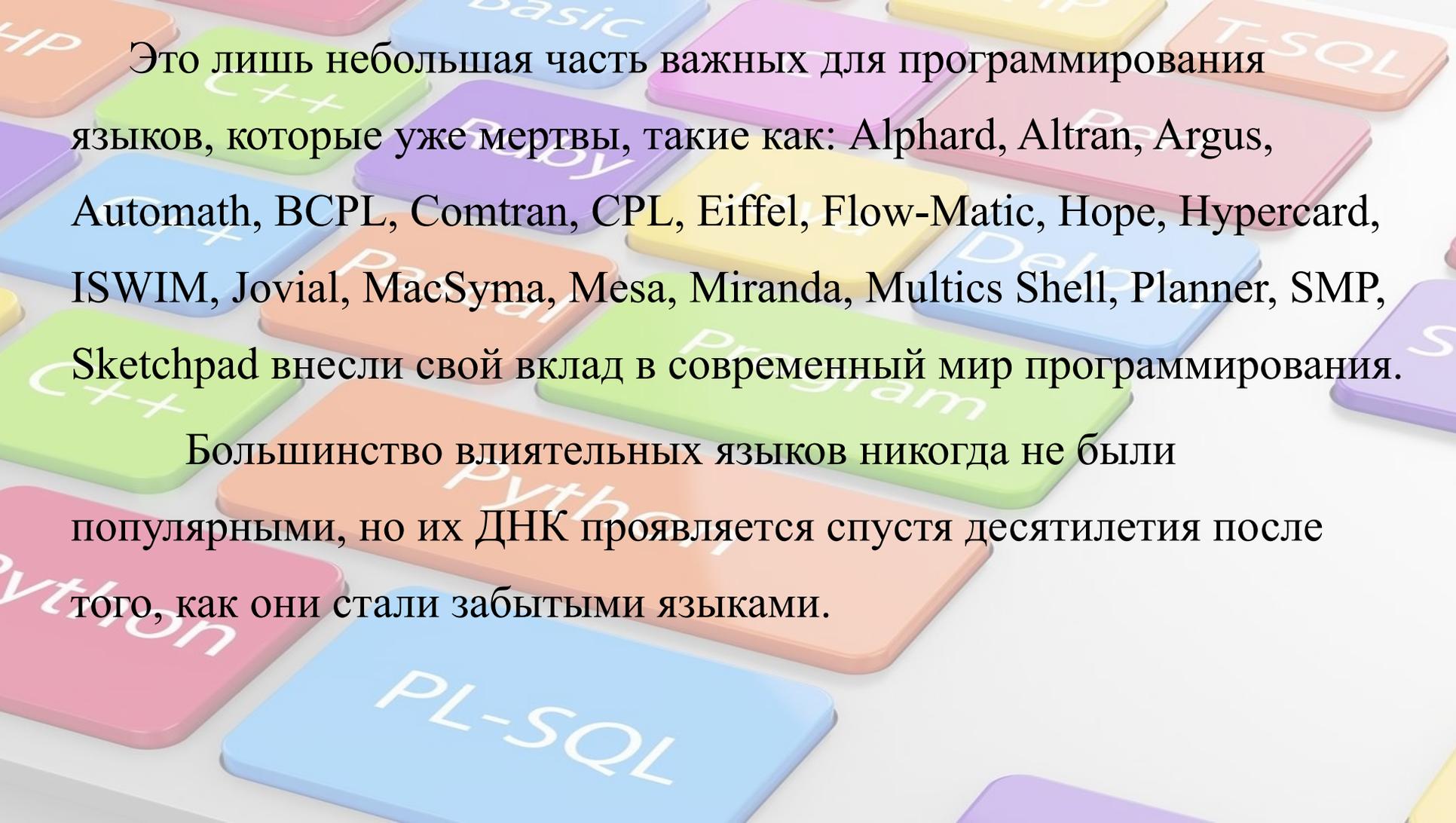
Разработчики: Оле Даль и Кристен Найгаард, 1967.

Simula 67 — расширенная версия Algol для математического моделирования. Первая версия языка (Simula I) имела специальный синтаксис моделирования — разработчикам показалось, что он получился слишком специализированным, а в симуляциях содержалось слишком много дублирования кода. Даль и Найгаард хотели создать более универсальный язык, возможности которого выходили бы за пределы моделирования.

Их идея заключалась в том, чтобы дать пользователям возможность определять новые типы объектов — классы — с разрешением полиморфного определения функций. После этого пользователи могли создать функции моделирования как частные случаи объектно-ориентированной системы.

Задача	Примеры языков
Задачи искусственного интеллекта	Lisp, Prolog, Multilisp, Commonlisp, Рефал, Planner, QA4, FRL, KRL, QLisp
Параллельные вычисления	Fun, Apl, Alfl, PARAlfl, ML, SML, PPL/1, Hope, Miranda, Occam, PFOR, Glypnir, Actus, параллельный Cobol, ОВС-ЛЯПИС, ОВС-Мнемокод, ОВС-Алгол, ОВС-Фортран, PA(1), PA(G)
Задачи вычислительной математики и физики	Occam, PFOR, Glypnir, Actus, параллельный Cobol, ОВС-ЛЯПИС, ОВС-Мнемокод, ОВС-Алгол, ОВС-Фортран, PA(1), PA(G)
Разработка интерфейса	Forth, c, C++, Ассемблер, Макроассемблер, Simula-67, ОАК, Smalltalk, Java, РПГ
Разработка программ-оболочек, разработка систем	Forth, c, C++, Ассемблер, Макроассемблер, Simula-67, ОАК, Smalltalk, Java, РПГ
Задачи вычислительного характера	Algol, Fortran, Cobol, Ada, PL/1, Фокал, Basic, Pascal

В данной таблице показаны задачи языков программирования.



Это лишь небольшая часть важных для программирования языков, которые уже мертвы, такие как: Alphard, Altran, Argus, Automath, BCPL, Comtran, CPL, Eiffel, Flow-Matic, Hope, Hypercard, ISWIM, Jovial, MacSyma, Mesa, Miranda, Multics Shell, Planner, SMP, Sketchpad внесли свой вклад в современный мир программирования.

Большинство влиятельных языков никогда не были популярными, но их ДНК проявляется спустя десятилетия после того, как они стали забытыми языками.

Список литературы

1. [картинки связанные с языками программирования simula 67: 1 тыс изображений найдено в Яндекс.Картинках \(yandex.ru\)](#)
2. [картинки связанные с языками программирования simula 67: 1 тыс изображений найдено в Яндекс.Картинках \(yandex.ru\)](#)
3. [картинки связанные с языками программирования simula 67: 1 тыс изображений найдено в Яндекс.Картинках \(yandex.ru\)](#)