

Языки и системы программирования

Обзор классификаций языков программирования

Симоненко Е.А.

Кубанский государственный технологический университет

2019

По областям применения

По способу написания программ

Императивный и декларативный подходы

Компилируемые и интерпретируемые языки

Низкоуровневые и высокоуровневые языки

- Языки для научных приложений
- Языки для коммерческих приложений
- Языки для искусственного интеллекта
- Языки системного программирования
- Языки сценариев (скриптов)
- Специализированные языки

Языки для научных приложений

- Fortran
- C, C++
- R
- Python, Julia
- Scilab, Maxima, GNU Octave, Axiom, Reduce
- Haskell, Agda

Языки для коммерческих приложений

- Java, C#
- C, C++
- Perl, Python, PHP
- Haskell
- JavaScript, TypeScript
- Lisp, Clojure, ClojureScript

- Prolog
- Python
- Lisp
- CLIPS

Языки системного программирования

- C, C++
- Rust
- Java

Языки сценариев (скриптов)

- Shell
- Perl
- Python
- JavaScript
- Emacs Lisp
- Scheme
- R

- Scratch
- Siemens Step 7
- Coq, Isabelle
- Ant
- OpenCL, CUDA: C-подобный язык для графических карт.

По способу написания программ

- текстовые
- графические

Подавляющее большинство языков программирования.
C++, Java, Python, C#, R ...

Редкий случай.

- Scratch: для обучения детей программированию.
- Siemens Step 7: для программирования PLC.
- UML: транслируется в «обычный» язык программирования.

- C, C++
- Fortran, Pascal
- Java, C#
- Python, Perl, R

- Prolog
- Haskell, Agda, Idris
- SQL

Система программирования содержит компилятор. Этап компиляции присутствует в явном виде: чтобы запустить программу, её нужно сначала скомпилировать.

- C, C++, Rust
- Fortran, Pascal
- Java, C#
- Haskell, Agda, Idris
- TypeScript
- Visual Prolog

Система программирования может содержать компилятор, но для запуска программы не требуется вызывать его в явном виде.

- Shell
- Perl, Python, R, Ruby
- JavaScript
- Scheme, ClojureScript, Emacs Lisp
- SWI-Prolog, GNU Prolog

Языки с компиляцией в машинный код

- C, C++, Rust
- Fortran, Pascal

Языки с компиляцией в байт-код

- Java, C#
- Haskell

Языки с компиляцией в другой язык

- TypeScript -> JavaScript
- Idris -> JavaScript

Интерпретируемые языки со «скрытой» компиляцией в байт-код

- Python, Ruby
- Perl
- R
- JavaScript
- Scheme, Emacs Lisp

«Подлинно» интерпретируемые языки

- Shell
- Scilab, Maxima, GNU Octave?
- SWI-Prolog, GNU Prolog?

Низкоуровневые и высокоуровневые языки

Когда-то языки более высокого уровня чем ассемблер считались высокоуровневыми. При таком подходе практически все распространённые языки следует считать высокоуровневыми. Более современный подход таков:

- *низкоуровневые языки* оперируют объектами непосредственно связанными с архитектурой компьютера и его операционной системой, как-то: байты, биты, адреса, прерывания, системные вызовы, потоки и процессы. Программы на таких языках как-правило компилируются в непосредственно исполняемый машинный код.
- *высокоуровневые языки* работают с абстракциями, представляющими математические или материальные объекты, например, с текстами, графами, виджетами, таблицами и записями баз данных. Такие языки обычно не имеют прямого доступа к процессору и памяти. Программы на них как-правило компилируются в байт-код или интерпретируются.

Низкоуровневые языки

- C
- C++
- Rust

- Prolog
- Haskell, Agda, Idris
- Python, R, Perl, Ruby
- Emacs Lisp, Scheme, ClojureScript
- Java, C#, Clojure

Благодарю за внимание!
Языки и системы программирования
Обзор классификаций языков программирования
Симоненко Е.А.
easimonenko@mail.ru