Програмни езици (Обектно-ориентирано програмиране)

Информация

- Аделина Алексиева-Петрова
- Email: aaleksieva@tu-sofia.bg
- Кабинет: 3311А

- Приемно време: вторник - 11:30

Програмни езици

- **Модел на изчисление:** стил на изграждане и елементите на програма

- **Изразителност**
  - Тип данни и техните оператори
  - Структури за контрол
  - Механизъм на абстракция
  - Инструменти

- **Леснота за използване:** Четливост/ Поддръжка

Модел на изчисление

- **Императивни**: промяна на състоянието на данните чрез команди в програмен код.
- **Функционално**: изчисление се базира на математически функции и се избягва промяна на състоянието на данните.
  - Lisp, Scheme, Haskell, F#
- **Декларативни**: твърдения и правила
  - Prolog, data base applications

The Generations of Programming Languages (1/2)

- First-generation languages (1954–1958)
  - FORTRAN I Mathematical expressions
  - ALGOL 58 Mathematical expressions
  - Flowmatic Mathematical expressions
  - IPL V Mathematical expressions
- Second-generation languages (1959–1961)
  - FORTRAN II Subroutines, separate compilation
  - ALGOL 60 Block structure, data types
  - COBOL Data description, file handling
  - Lisp List processing, pointers, garbage collection

The Generations of Programming Languages (2/2)

- Third-generation languages (1962–1970)
  - PL/1 FORTRAN + ALGOL + COBOL
  - ALGOL 68 Rigorous successor to ALGOL 60
  - Pascal Simple successor to ALGOL 60
  - Simula Classes, data abstraction
- The generation gap (1970–1980).

Object-orientation boom

- 1980–1990, but few languages survive
  - Smalltalk 80 Pure object-oriented language
  - C++ Derived from C and Simula
  - Ada83 Strong typing; heavy Pascal influence
  - Eiffel Derived from Ada and Simula

Emergence of frameworks (1990–today)

- Visual Basic - eased development of the graphical user interface (GUI) for Windows applications
- Java - successor to Oak; designed for portability
- Python - object-oriented scripting language
- J2EE - Java-based framework for enterprise computing
- .NET - Microsoft's object-based framework
  - Visual C# - Java competitor for the Microsoft .NET Framework
  - Visual Basic .NET - Visual Basic for the Microsoft .NET Framework

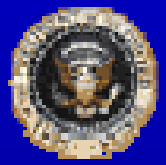
Common Ideas

- Modern imperative languages (C#, C++, Java) have similar characteristics:
  - large number of features (grammar with several hundred productions, 500 page reference manuals…)
  - a rich type system
  - procedural mechanisms
  - object-oriented facilities
  - abstraction mechanisms, with information hiding
  - several storage-allocation mechanisms
  - facilities for concurrent programming
  - facilities for generic programming

Predictable performance vs. ease of writing

- Low-level languages mirror the physical machine:
  - Assembly, C, Fortran
- High-level languages model an abstract machine with useful capabilities:
  - ML, Setl, Prolog, Python
- Wide-spectrum languages try to do both, more or less well:
  - Ada, C++, Java
- High-level languages are often interpreted, have garbage collector. Cost of operations is not directly visible.
  - Java is a hybrid

High-level Abstract Description to Low-level Implementation Details



President

My poll ratings are low, lets invade a small nation

General

Cross the river and take defensive positions

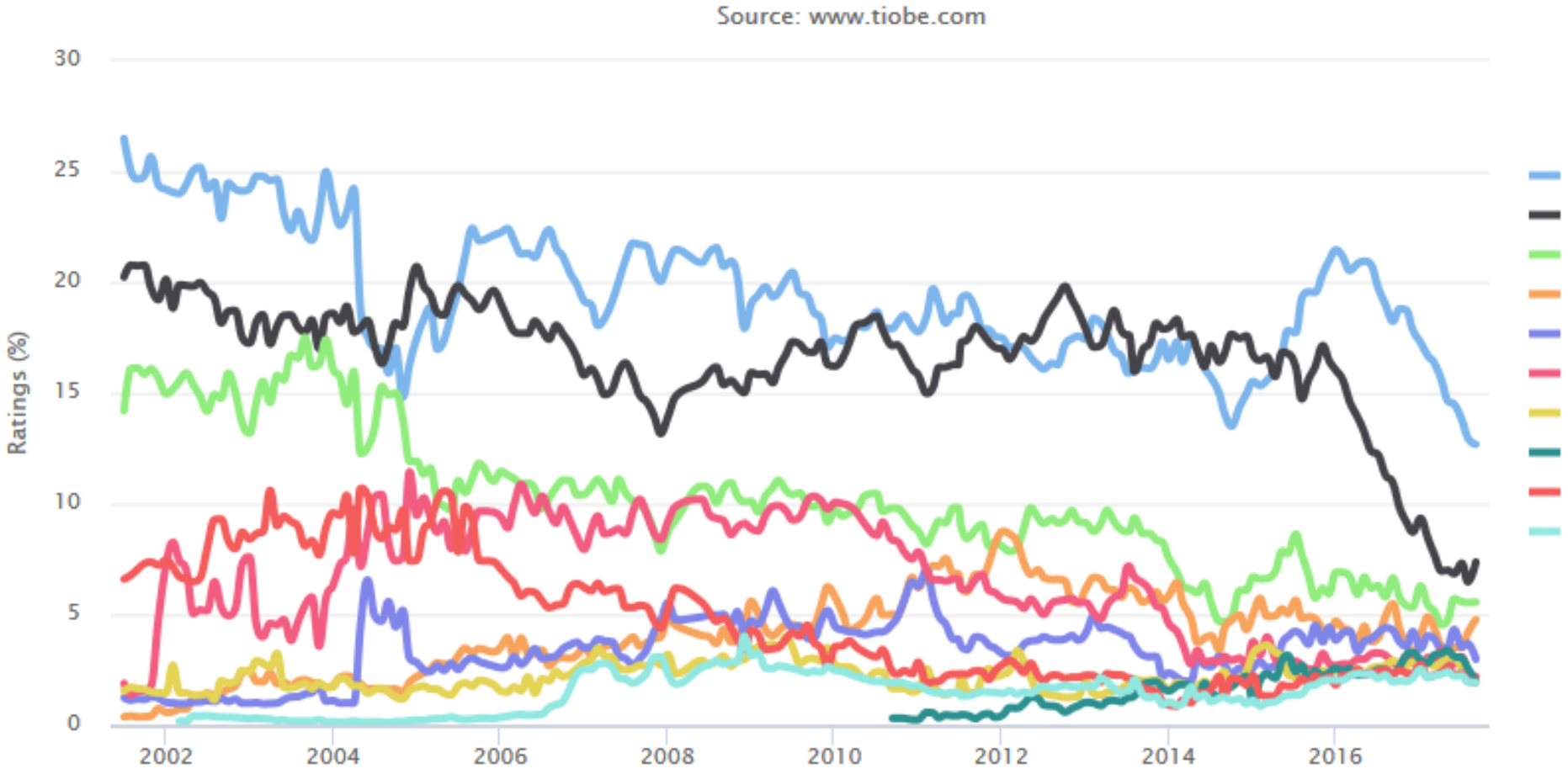
Sergeant

Forward march, turn left Stop!, Shoot

Foot Soldier

TIOBE Index for September 2017

Sep 2017	Sep 2016	Change	Programming Language	Ratings	Change
1	1		Java	12.687%	-5.55%
2	2		C	7.382%	-3.57%
3	3		C++	5.565%	-1.09%
4	4		C#	4.779%	-0.71%
5	5		Python	2.983%	-1.32%
6	7	▲	PHP	2.210%	-0.64%
7	6	▼	JavaScript	2.017%	-0.91%
8	9	▲	Visual Basic .NET	1.982%	-0.36%
9	10	▲	Perl	1.952%	-0.38%
10	12	▲	Ruby	1.933%	-0.03%


Source: www.tiobe.com
Ratings (%)
30
25
20
15
10
5
0
2002
2004
2006
2008
2010
2012
2014
2016