Tales from the workshops

Dr Russel Winder

email: russel@winder.org.uk
twitter: @russel_winder
web: http://www.russel.org.uk
Agile is the new black.
Aims, goals, objectives...
Aims, goals, objectives...

- Explore some experiences teaching new programming languages to programmers.
- Try and find some signposts to PPIG type things.
Agile is passé?
Structure

Beginning
Middle
End
Personal Path

FORTRAN
Assembly language
Algol-68
Pascal
C

Prolog
Scheme
C++
Miranda
Java
Fortran
Haskell

Modula-2
Java
Python
Clean

Groovy
D
Scala

Fortress
X10
Chapel
C#
Clojure
Go
Ceylon
Kotlin
Rust
Frege
The university Years
Dungeons and Dragons

- AD&D obviously, not D&D.
- Student projects often in C, not a good language for writing these games and tools.

Pascal past its "use by" date.
Lisp might have been useful.
MMORPG

- MUD, and the like...
- Game description languages...
- Domain Specific Language (DSLs) especially internal rather than external.

It's all about abstraction, tokens, and programming activity.
e.g. Python

- For:
  - Bounded iteration
- while:
  - Unbounded iteration

- Functions:
  - Activity token
- Classes:
  - Object factory.

It’s all about abstraction, tokens, and programming activity.
BCS HCI SG

- Humans as part of systems.
- Creating good uis, with good ux.
- Software development tools and the coming of IDEs.

As opposed to UNIX philosophy of OS, shell, and tools.
undergraduates

- Polyglot approach:
  - Scheme then C++
  - Miranda then C++

- Monoculture:
  - Java

wot no Python?

The role of REPLs.
Postgraduates

- Mentor do not supervise.
- Philosophy matters.
- Computational models, operational semantics, and declarative semantics are critical.
A Language war

Functional vs. object-oriented
The Start-up Years
Languages

- Realizing a novel low-level machine.
- C, (C++), Java.
- Python for driving integration and system tests of the C codes.
Mentor not Manage

- Self-directing teams.

Best thinkers/software designers not always the best programmers.

Agile before Agile.
O
The Last 10 Years
The workshops

- Introducing a programming language to programmers used to other programming languages.
- Taking programmers who “sort of” know a programming language much further into that programming language.
Introduction to topic with relevant examples.

Pair working on some related problems.

Mob working or leader led live coding.
Adults...

- ...do not feel playing with things is right.
- ...feel they have to construct without error rather than try and (fail | succeed).
- ...worry they should not have fun.

Happy, smiling people tend to learn more?
Problems...

- ...small enough to be completed; or
- ...small changes in a big code base.

Development vs. Maintenance.
Programmers...

- ...cannot seem to do as much as they think they can in 40mins.
- ...seem not to know as much as they (think I should).
Pairing...

- ...can be hard to get programmers to do.
- ...increases the energy in workshops.
- ...leads to more fun, and less stress about getting things wrong.

Happy, smiling people tend to learn more?
Feedback...

- ...is crucial to learning.
- ...must be integral to workshop structure, both individual and group.
Introduction to topic with relevant examples.

Pair working on some related problems.

Mob working or leader led live coding.
A Language war

Statically typed vs. Dynamically typed
Static languages...

- ...compiler support for type checking.
- ...compiler generated, ahead of time optimized code.
- ...strong IDE support.
Dynamic Languages...

- ...interpreted or bytecode compiled.
- ...slower (but...).
- ...hard for IDE to give excellent support.
- ...generally use "duck typing".
Dynamic Objects

- An object has no "defined for all runtime" interface.

- An object's interface can only be determined at run time, and may change immediately after being checked.
Duck Typing

- If an object has a particular method at the moment of being asked to execute that method, then the object must be of the right type.
The Points...

- ...some programmers, used to statically typed languages, cannot cope with dynamically typed ones.

- ...IDEs have a hard time giving programmers the support expected from the statically type language experience.
what is the value of $\pi$?
It's simples.
Programmers...

• ...generally do not know as much about floating point hardware as they (think I should).
Programmers...

• ...generally know less about concurrency and parallelism that they (think I should).
IDEs...

- ...make development much easier for some.
- ...definitely work better with static languages.
- ...appear to be enforcing static typing.
Python

- Even the doyen of dynamic languages is investigating type hinting, cf. PEP 484 and MyPy.
Groovy...

- ...is an optionally typed language.
- Can use it fully dynamic or with run time checked types.
- Or with annotations, as a statically type checked, and even compiled, language.
Tales from the workshops

Dr Russel Winder

e-mail: russel@winder.org.uk

twitter: @russel_winder

web: http://www.russel.org.uk