

*Welche Programmiersprache
unterstützt meine
Konzepte für den
Informatikunterricht?*

Ludger Humbert

22. Juli 2002 – 21:53 Uhr

Konzepte

Wechselwirkung zwischen Konzepten und Sprachen

Informatische Modellierung ist allgemein bildend

fachwissenschaftlicher Hintergrund

Übergang von monolithischen Programmsystemen zu komplexen, miteinander vernetzten und interagierenden Informatiksystemen

didaktische Gestaltung

Sicht sowohl auf die verschiedenen Phasen des Software-Lebenszyklus, aber auch auf die Entwicklung von interaktiven Benutzungsschnittstellen und der damit verbundenen notwendigen Benutzerorientierung

Sprache = Lernhilfe?

im Informatikunterricht eingesetzten Sprachen sind für professionelle Entwicklerinnen entwickelt worden

- ⇒ Entwicklungsziel bei Computersprachen:
Unterstützung professioneller Softwareentwicklung
 - Voraussetzung und Zieldimension, die für Schülerinnen nur begrenzt zutreffen
 - Sprachen sind nicht zum Einsatz zur Unterstützung von Lehr-/Lernprozessen konzipiert
- ⇒ Deshalb sind professionelle Sprachen für die Eignung im Lehr-/Lernprozess sorgfältig zu prüfen.

Schnelle Softwareentwicklung

Interessen

professioneller
Entwickler

Schüler,
Ausbilder

GUI-Entwurf
Rapid Prototyping
Generatoren
Bibliotheken
Dokumentation
Wiederverwendung
...

Portabilität
Teamarbeit
Lesbarkeit
Fehlertoleranz
...

Grundlagen
Erlernbarkeit
Algorithmen
Leichte Syntax
Orthogonalität
...

?

Gemeinsamkeiten?
Konflikte?

Zielsetzung

Kategorien zur Auswahl von Sprachen

A. *fachwissenschaftlicher Bereich*

Informatische Modellierung

B. *lerntheoretischer Hintergrund*

konstruktivistische Sicht auf Lehr-/Lernprozesse – verbunden mit einer handlungsorientierten Didaktik

C. *Zielorientierung*

didaktische Gestaltung – frühzeitiges Prototyping, Diskussion mit zukünftigen Nutzerinnen, ausgestaltet auf einer lernförderlichen Basis

Konzepte

Lernhilfen

Kategorien

Modellierung

Fallstudie

Folgerungen

Webseite

Erste



Seite 5 von 11

Zurück

Vollbild

Schließen

Beenden

Modellierung

alle Modellierungsvarianten sollten(!) von der gewählten Sprache unterstützt werden

⇒ imperativ – von Neumann Sprache(n)

didaktische Erkenntnisse liegen vor

⇒ objektorientiert

zunehmende Erfahrungen

⇒ funktional

viele singuläre Erfahrungsberichte

⇒ wissensbasiert

Annahmen über „die Welt“

⇒ Python?

PyNassi

Fallstudie

PyNassi: Gestaltung einer explorativen Umgebung

↪ in Python realisierte Applikation, die auf einer interaktiven Basis gestattet:

- ➔ Struktogramme zu erstellen,
 - ➔ aus einem Struktogramm Python-Quellcode zu erzeugen,
 - ➔ Abläufe
 - im Struktogramm,
 - auf der Ebene des Quellcodes,
 - und auf der Aktionsebene
- zu verfolgen und dynamisch zu visualisieren.

[Freeware] - PyNassi 0.4 von Inli, Uni DO

Datei Ablauf Hilfe

Struktogramm (D:/Eigene Dat...)

```

graph TD
    subgraph "Funktion fakt(x)"
        direction TB
        A["if x>0"] -- then --> B["return x*fakt(x-1)"]
        A -- else --> C["return 1"]
    end
    subgraph "Funktion main()"
        D["print fakt(5)"]
        E["main()"]
    end
  
```

Source (noname.py)

```

def fakt(x):
    if x>0:
        return x*fakt(x-1)
    else:
        return 1
def main():
    print fakt(5)
main()
  
```

Textausgabe (stdou...)

```

Hallo \Welt
120
120
  
```

Programm-Überwachung

- 1 if x>0: {x: 1} / None
- 2 return x*fak {x: 1} / None
- 0 def fakt(x): {x: 0} / None
- 0 def fakt(x): {x: 0} / None
- 1 if x>0: {x: 0} / None
- 4 return 1 {x: 0} / None
- 4 return 1 {x: 0} / 1
- 2 return x*fak {x: 1} / 1
- 2 return x*fak {x: 2} / 2
- 2 return x*fak {x: 3} / 6
- 2 return x*fak {x: 4} / 24
- 2 return x*fak {x: 5} / 120
- 6 print fakt(5) () / None
- 3 exit

Willkommen im PyNassi Struktogrammeditor!

OK

Abbildung „PyNassi“

Maßgaben?

kleine, spezialisierte und im Quellcode erhältliche Werkzeuge

Untersützung der Modellierung

- ⇒ dia
- ⇒ dia2code
- ⇒ BoaConstructor
- ⇒ PythonCard
- ⇒ ...

Konzepte

Lernhilfen

Kategorien

Modellierung

Fallstudie

Folgerungen

Webseite

Erste

◀▶

◀▶

Seite 9 von 11

Zurück

Vollbild

Schließen

Beenden

Hinweise auf Materialien – URLs

→ **Masters thesis:**

"How applicable is Python as first computer language for teaching programming in a pre-university educational environment, from a teacher's point of view?"

<http://amstel.science.uva.nl/~fotisg/python/>

→ **PyNassi und die komplette Diplomarbeit (nach Fertigstellung):**

„Eignet sich die Skriptsprache Python für schnelle Entwicklungen im Softwareentwicklungsprozess?“

(ab Ende August 2002):

<http://www.ingo-linkweiler.de/diplom/>

Webseite

Erste



Seite 10 von 11

Zurück

Vollbild

Schließen

Beenden

Konzepte

Lernhilfen

Kategorien

Modellierung

Fallstudie

Folgerungen

Webseite

Erste



Seite 11 von 11

Zurück

Vollbild

Schließen

Beenden