Unterricht planen

Fachdidaktisches Seminar I & II

SS 2018

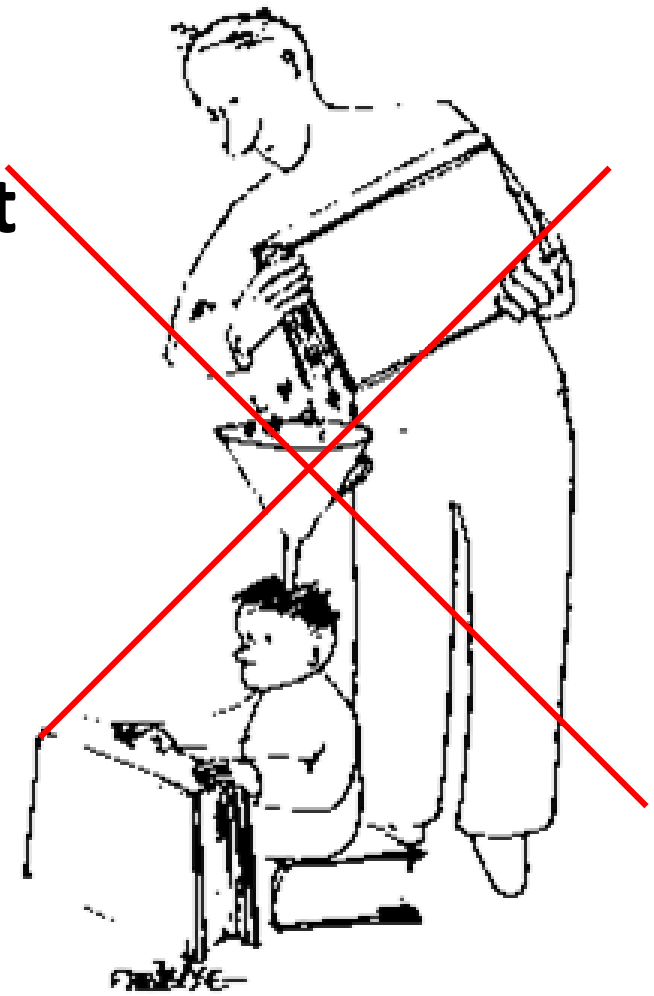
Univ.-Prof. MMag. Dr. Barbara Sabitzer
School of Education – MINT-Didaktik
Johannes Kepler Universität Linz
barbara.sabitzer@jku.at

JOHANNES KEPLER UNIVERSITÄT LINZ

Das Wichtigste zuerst

Wissen kann **nicht**
übertragen oder **vermittelt**
werden – es muss vom
Lernenden
neu geschaffen werden.

Lernen ist immer ein
aktiver und **selbstgesteuerter**
Prozess



Neurodidaktik Schlagzeilen

- **Use it or lose it!**
- Automatische **Mustererkennung** & **Regelextraktion**
- Das Gehirn ist ein **Spiegel.**
- Doppelt hält besser! → **Multimediaeffekt**
- „Lernen im Schlaf“ → **Konsolidierung**
- Lernen MUSS **Sinn** machen → Bezug, Nutzen
- 4 Augen sehen mehr als 2 – **Kooperation**
- Neugier + Entdecken = **Dopamin** → **Belohnung**
- Neurodidaktik bestätigt **Reformpädagogik**

Gehirngerecht Lehren und Lernen (1/2)

(1) Biologische Faktoren beachten
(Alter, Geschlecht, Hormone, Neurotransmitter)

→ Wahlmöglichkeiten bieten
- Themen
- Lehr- und Lernmethoden
- Art und Formulierung der Aufgaben
- Lernrhythmus
- Sozialform etc.

(2) Rahmenbedingungen gestalten
(LV-Organisation, Material, Kooperation, Ort, Zeit)

→ Offener Unterricht
- COOL
- EVA-Lernen
- SchülerInnen in Entscheidungen einbeziehen

(3) Individuelle Faktoren beeinflussen
(Interesse, Motivation, Emotionen, Aufmerksamkeit etc.)

→ Wahlfreiheit

Gehirngerecht Lehren und Lernen (2/2)

(4) Gehirn- und Gedächtnisfunktionen unterstützen

→ kognitive Lerneffekte nutzen

- Automatische Mustererkennung
  → Entdeckendes Lernen, Schritt-für-Schritt-Aufgaben, Musterlösungen
- Konsolidierung
  → Pausen, Schlaf
- Spiegelneuronen
  → Lernen am Modell, Schritt-für-Schritt-Aufgaben
- Arbeitsgedächtnis
  → Lernstrategien, Kognitive Belastung reduzieren
- Priming
  → Modellierung, Concept Maps, MindMaps
- Abruf = Neueinspeicherung
  → Lernen durch Lehren, Gruppenpuzzle, Pair-Programming
- Effekte
  → Priming, Primacy-Recency, Multimedia

Lesson & Task Design

- Lesson Design
  - Rahmenbedingungen
  - Ziele
  - Unterrichtskonzept und -methoden
  - Unterrichtsverlauf
- Task Design
  - Vorzeigematerialien
  - Musterbeispiele und -lösungen
  - Aufgabenstellungen
  - Arbeitsaufträge
  - Literatur

Unterrichtskonzepte – Lesson Design

- COOL http://cooltrainers.at/
  - Freedom - Wahlfreiheit und Eigenverantwortung
  - Co-operation – Zusammenarbeit und Teamfähigkeit
  - Budgeting time - selbstständiges Planen und Organisieren
- COOL Informatics
  - Discovery
  - Cooperation
  - Individuality
  - Activity

COOL Informatics – Overview

Principle	Teaching and learning methods	Neurodidactical basis
1. Discovery	Solution-based learning (worked examples) Step-by-step instructions + tasks Video tutorials, Observational learning Learning with all senses	Pattern recognition Mirror neurons Individual learning rhythm modality / multimedia effect
2. Cooperation	Team and group work Peer tutoring and -teaching Pair programming Cross-curricular learning Project-based learning	"A joy (=knowledge) shared is a joy (=knowledge) doubled." Recall = re-storage in memory Integrating individual needs, talents, competences, practical relevance
3. Individuality	Competence-based learning Questioning Self-organized learning with compulsory and optional tasks	Connecting new information to previous knowledge, Considering individual interests, needs, tasks, methods, learning rhythm
4. Activity	Hands-on, Minds-on Learning by doing Learning by animation, simulation by playing and designing games (creative learning)	Knowledge must be newly created (constructed) by each learner (= constructivism) Learning is an active process (=progressive education)

Brain-based Programming LV-Organisation

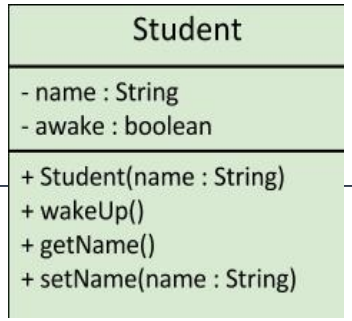
- Gruppen nach Kompetenzniveau
  - Profis = Peer-Tutoren, Peer-Teacher
  - Amateure = teilweise Peer-Tutoren
  - Einsteiger
- LV-Phasen (nicht unbedingt in dieser Reihenfolge)
  - Fragerunde (ca. 10 min)
    Vorwissen berücksichtigen, Abruf = Neueinspeicherung
  - (Vortrag bei Bedarf max. 20 min)
  - Entdecken (10-20 min)
    Mustererkennung, Lernrhythmus (Konsolidierung)
  - Labor (Pair-Programming)
    Practice makes perfect, Abruf = Neueinspeicherung

Brain-based Programming Aufgaben

- Entdeckendes Lernen - Mustererkennung
  - Leseecken (Programmcode, Puzzle, Lückentext, Fehlersuche)
  - Schritt-für-Schritt-Anleitungen mit Lösungen
  - Aufgabenstellungen mit Musterlösung
- Kompetenzorientiertes Üben – Vorwissen
  - Miniübungen
- Programmieren – Selbständiges Üben
  - Kleine, vollständige Programme
  - Teile bzw. Unterprogramme für ein Semesterthema

Leseecke

```
public class Student {

   private String name;
   private boolean awake;

   public Student(String name) {
      this.name = name;
      this.awake = false;
   }

   public String getName() {
      return this.name;
   }

   public void setName(String name) {
      this.name = name;
   }

   public void wakeUp() {
      this.awake = true;
   }
}
```




```
public class Studententag {

   public static void main(String[] args) {
      Student object1 = new Student("Jamie");
      Student andy = new Student("Andreas");
      Student randomName = new Student("Naomi");

      object1.wakeUp();
      andy.wakeUp();

      randomName.setName("Melanie");
      System.out.println(randomName.getName());
   }
}
```

8. Fragen und Aufgaben zur Leseecke 1

1. Lesen Sie die obigen Beispiele für Klassen und schreiben Sie einen „Schwindelzettel“ mit den wesentlichen Informationen zu Klassen und Objekten.
2. Markieren Sie den Konstruktor in der Klasse Student. Welche Parameter bekommt er, welche Variablen setzt er? Welche Methoden hat die Klasse Student? Was leisten sie?
3. Wie viele Objekte werden in der main-Methode (Klasse Studententag) erstellt? Wie heißen die Objekte?
4. Was wird an den Objekten verändert? Was wird schließlich auf der Konsole ausgegeben?

Schwierige & Komplexe Inhalte

- Verschiedene Perspektiven
- Verschiedene Aspekte
- Aussagekräftige Beispiele
- Musteraufgaben – Worked Examples
- Spiralförmig
  1. Minimalwissen
  2. Aufbau
  3. Ausnahmen
- Bezug zum Alltag bzw. bekannten Themen
- Für alle Sinne
- Zum BeGREIFen und ErLeben

Rekursion

Beispiel für eine Aufbereitung von
schwierigen und komplexen Themen

Konzept verstehen
Grundbegriffe kennenlernen
Einführung im Schulunterricht

JKU
JOHANNES KEPLER
UNIVERSITÄT LINZ

Rekursion – Stationen

Mögliche Stationen

- Sehen & Begreifen
- Sprache & Kunst
- Bewegung & Aktivität
- Mathematik & Geometrie
- Algorithmen & Programmieren
- Rekursion & Iteration
- Definitionen & Grundlagen

Prinzipien

COOL Informatics

- Discovery
- Cooperation
- Individuality
- Activity

Sehen & Begreifen (1/2)

- Eine Matrjoschka (ru. Матрёшка)

  ist eine Puppe,

  in der eine Puppe ist,

  in der eine Puppe ist,

  in der ...

- Ein Objekt ist rekursiv,
  - wenn es sich selbst enthält
  - oder durch sich selbst definiert ist.

Puppe in der Puppe

Sehen & Begreifen (2/2)

http://www.lamaisondelavachequirit.com/

Sprache & unendliche Rekursion

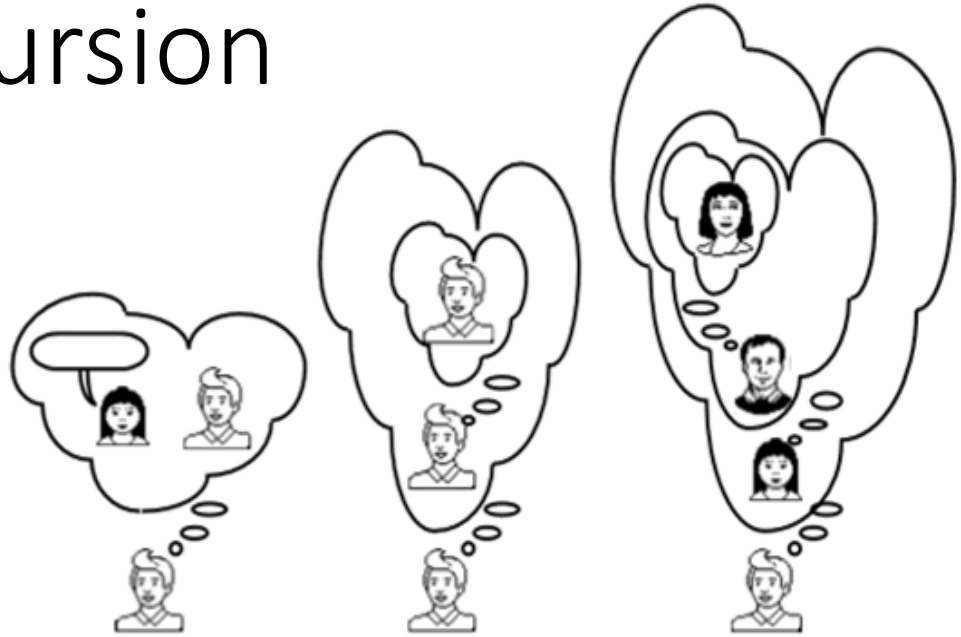
- Ein Lied:
  „Ein Hund kam in die Küche und stahl dem Koch ein Ei.
  Da nahm der Koch den Löffel und schlug den Hund zu Brei.
  Da kamen viele Hunde und gruben ihm ein Grab und
  bauten einen Grabstein, auf dem geschrieben stand:
  - „Ein Hund kam in die Küche ...

- Eine Geschichte:
  Es war einmal ein Mann, der hatte 7 Kinder, und
  die Kinder sprachen: „Vater, erzähl' uns eine Geschichte“. Da
  fing der Vater an:
  - „Es war einmal ein Mann, der hatte 7 Kinder ...

Sprache & Rekursion

- Rekursives Denken

  Max denkt,
  dass Karin denkt,
  dass Peter...



aus: Schwill, Andreas. "Ab wann kann man mit Kindern Informatik machen.“ *INFOS2001-9. GI-Fachtagung Informatik und SchuleGI-Edition* (2001): 13-30.

- Verschachtelte Sätze

Der Mann, heißt Peter.
der das Haus, gebaut hat,
das am Fluss, steht,
der durchs Tal fließt,

Rekursonsabstieg

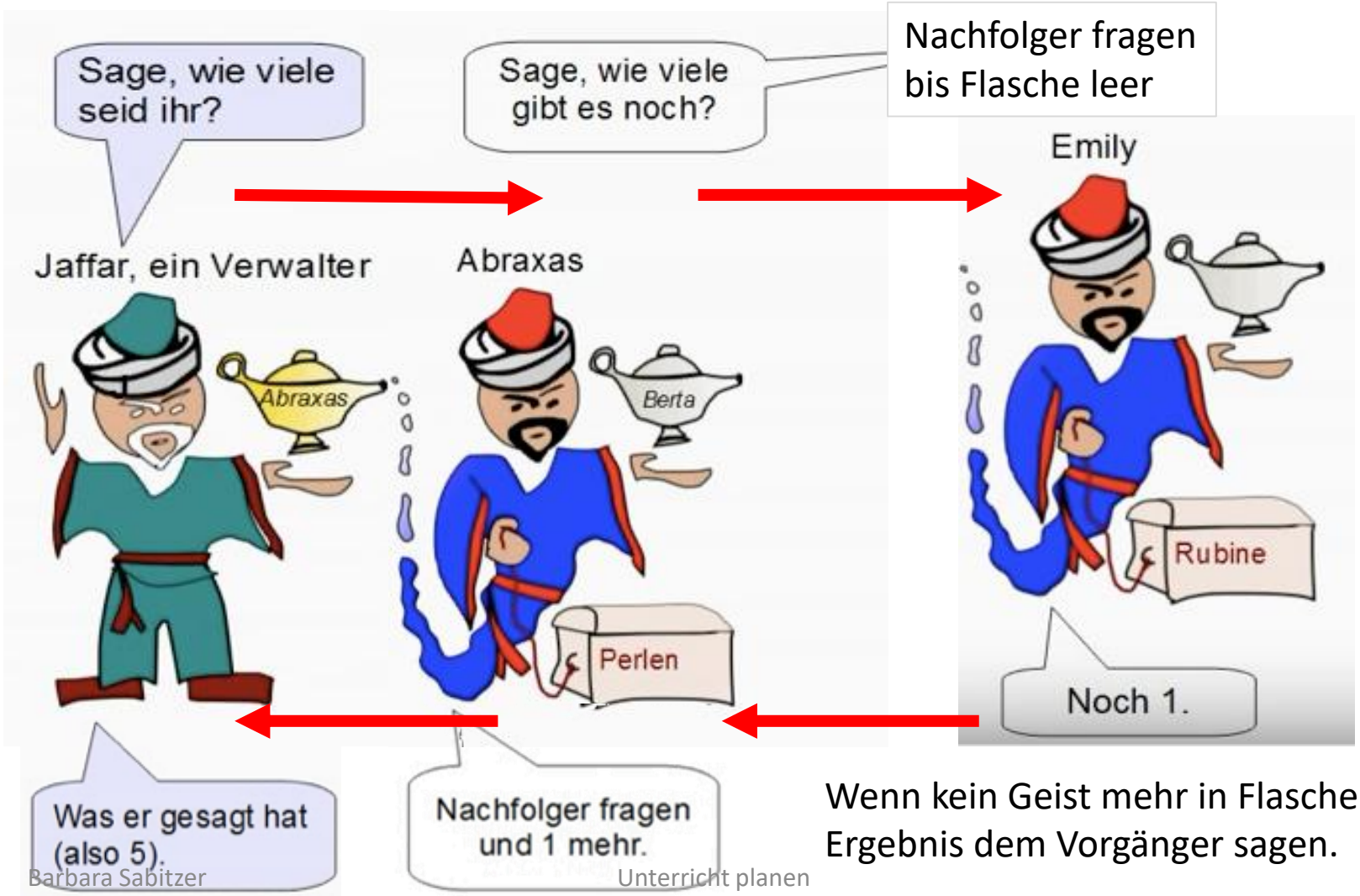
Rekursonsauftieg

Aufrufstruktur – Kompakte Darstellung sämtlicher rekursiver Aufrufe einer rekursiven Funktion/Methode

Rekursionstiefe – Anzahl der geschachtelten Aufrufe einer rekursiven Funktion/Methode

Rekursion – Verstehen

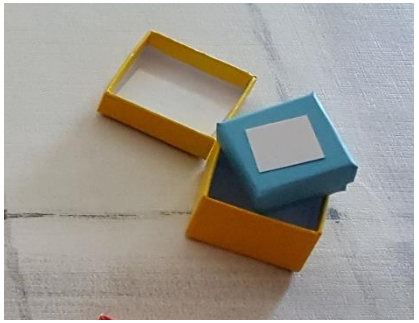
Jaffar & sein Flaschengeist https://www.youtube.com/watch?v=0kIPPQD19PM



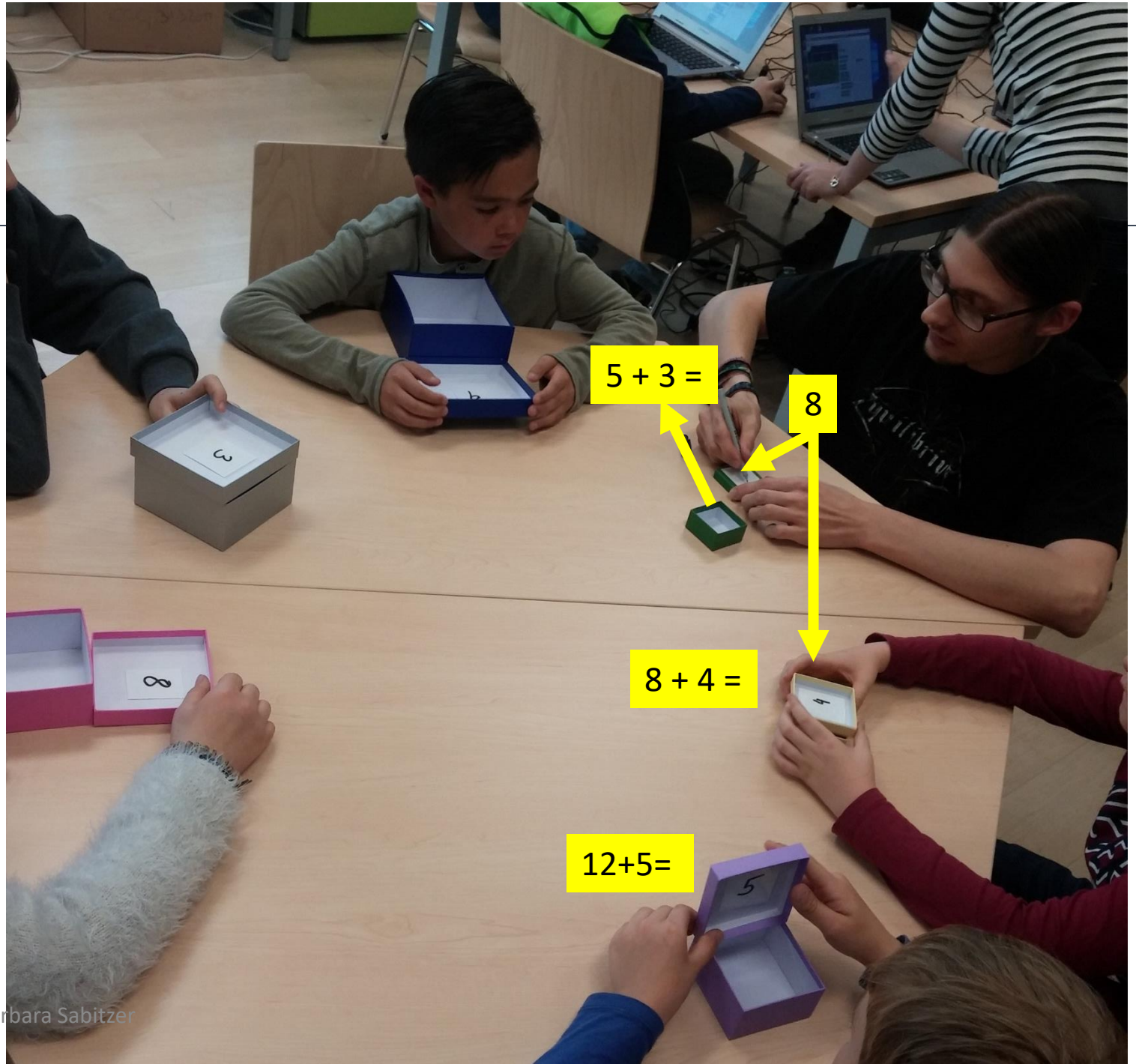
Wenn kein Geist mehr in Flasche: Ergebnis dem Vorgänger sagen.

Rekursion – Bewegung & Aktivität

- Wie viele gibt es?
  - Flaschengeist nachspielen
  - Matrjoschkas auspacken
  - Schachteln gesamt



- Schachtelrechnung
  - Rechnung in innerster Schachtel
  - In jedem Deckel
  - Endergebnis ganz außen


5 + 3 =
8
8 + 4 =
12+5=

Rekursion vs. Iteration

© J. Rau Januar 2013

Vergleich von Rekursion und Iteration

Iteration ist die Wiederholung strukturgleicher Blöcke durch **Aneinanderreihung**

Iteration (Endrekursion):
Man öffnet die größte Puppe, entnimmt die kleinere Puppe und setzt die beiden Teile der größeren Puppe wieder zusammen. Dies wiederholt man, bis die kleinste Puppe ausgepackt ist.

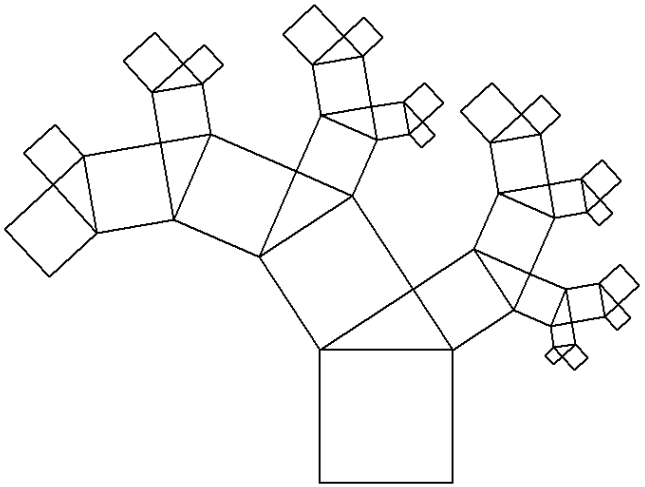
Rekursion ist die Wiederholung strukturgleicher Blöcke **durch Schachtelung**

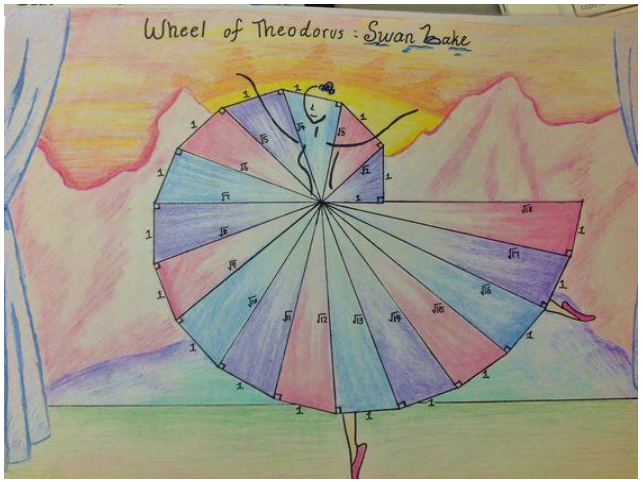
Echte Rekursion:
Man öffnet nacheinander jede Puppe, stellt deren 2 Teile geöffnet ab und entnimmt die nächste Puppe. Wenn die kleinste Puppe entnommen wurde, setzt man die jeweiligen zwei Teile der größeren Puppen in umgekehrter Reihenfolge zusammen.

http://www.j-rau.de/jogohegy/gk12/15RekursiveAlgorithmen.pdf

Mathematik & Geometrie (1/2)

- Geometrie
  - Der Baum des Pythagoras
  - Das Rad des Theodorus
  - Kochkurve
  - …

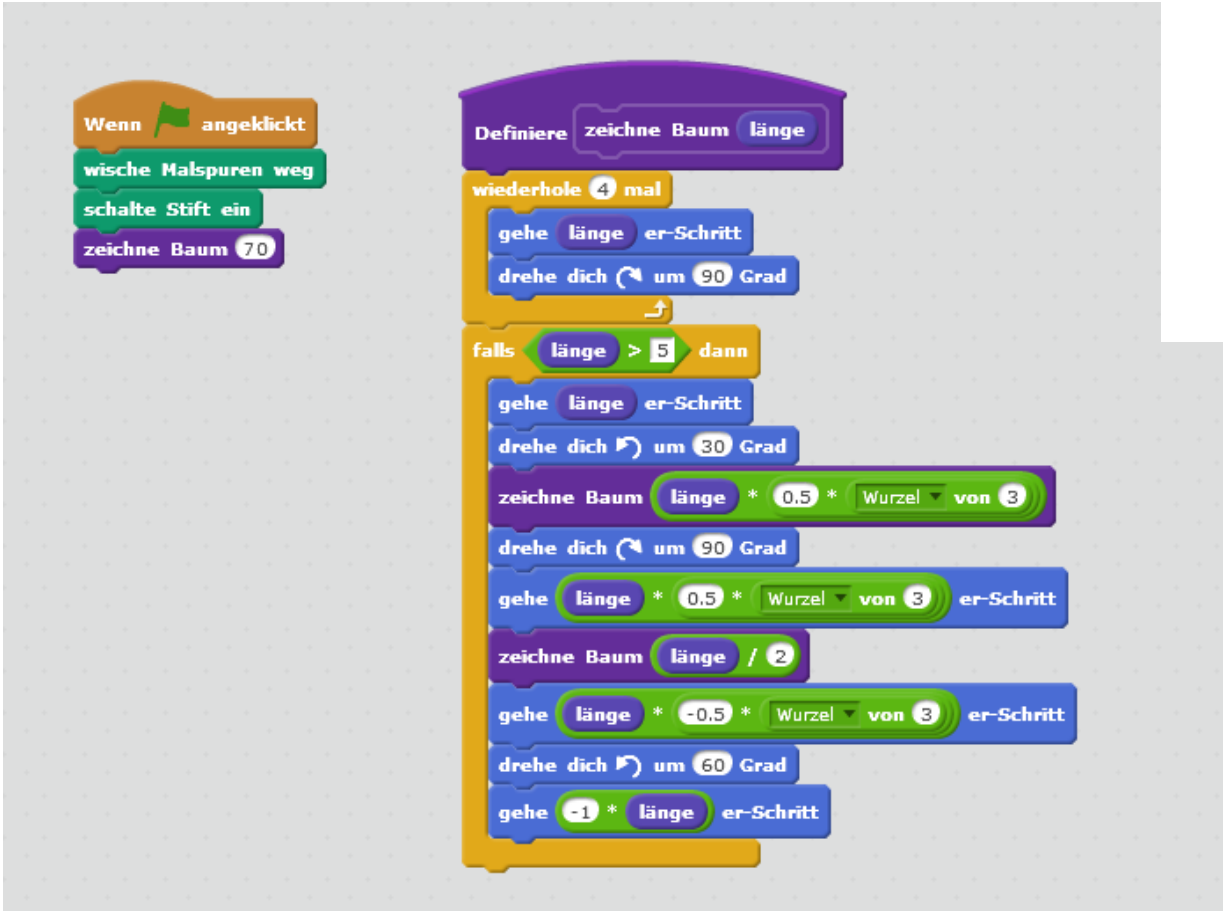


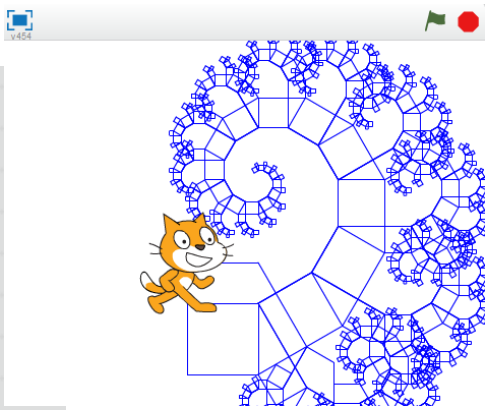


https://www.pinterest.com/pin/546694842236347302

- Mathematik
  - Intelligenztest: Zahlenreihen
  - Fibonacci
  - Fakultät
  - ggT

Mathematik & Geometrie (2/2)





Rekursive Algorithmen (1/4)

Grundstruktur

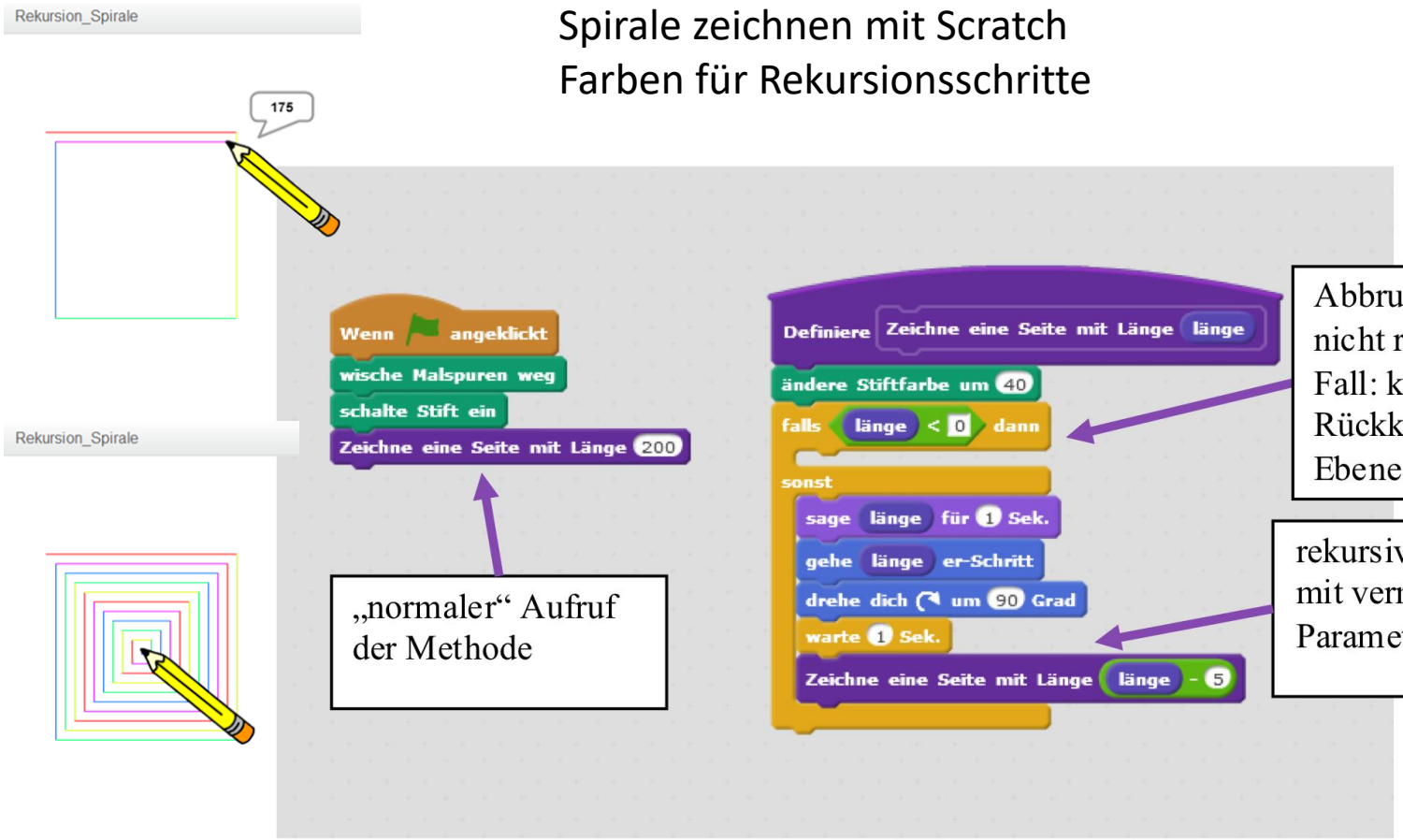
```
Programm Löse (Problem)
Begin
   //1. Rekursionsbasis im if = Abbruchbedingung
   If Problem einfach
            Löse Problem direkt
   //2. Rekursionsvorschrift
   Else
            Zerlege das Problem in kleinere, gleichartige Probleme
            Löse (kleines Problem)
            Verwende kleines Problem zur Lösung des Problems
   End
End
```

Puppen spielen mit Matrjoschkas

```
Programm Auspacken(Matrjoschka)
Begin
   If kleinste Matrjoschka gefunden
            Problem gelöst
   Else
            Öffne die Matrjoschka
      Entnimm die nächste Matrjoschka
            Auspacken(Matrjoschka)
   End
End
```

Algorithmen & Programmieren (2/4)

Spirale zeichnen mit Scratch
Farben für Rekursionsschritte



Algorithmen & Programmieren (3/4)

5.1 Leseecke

Zeit vergeht

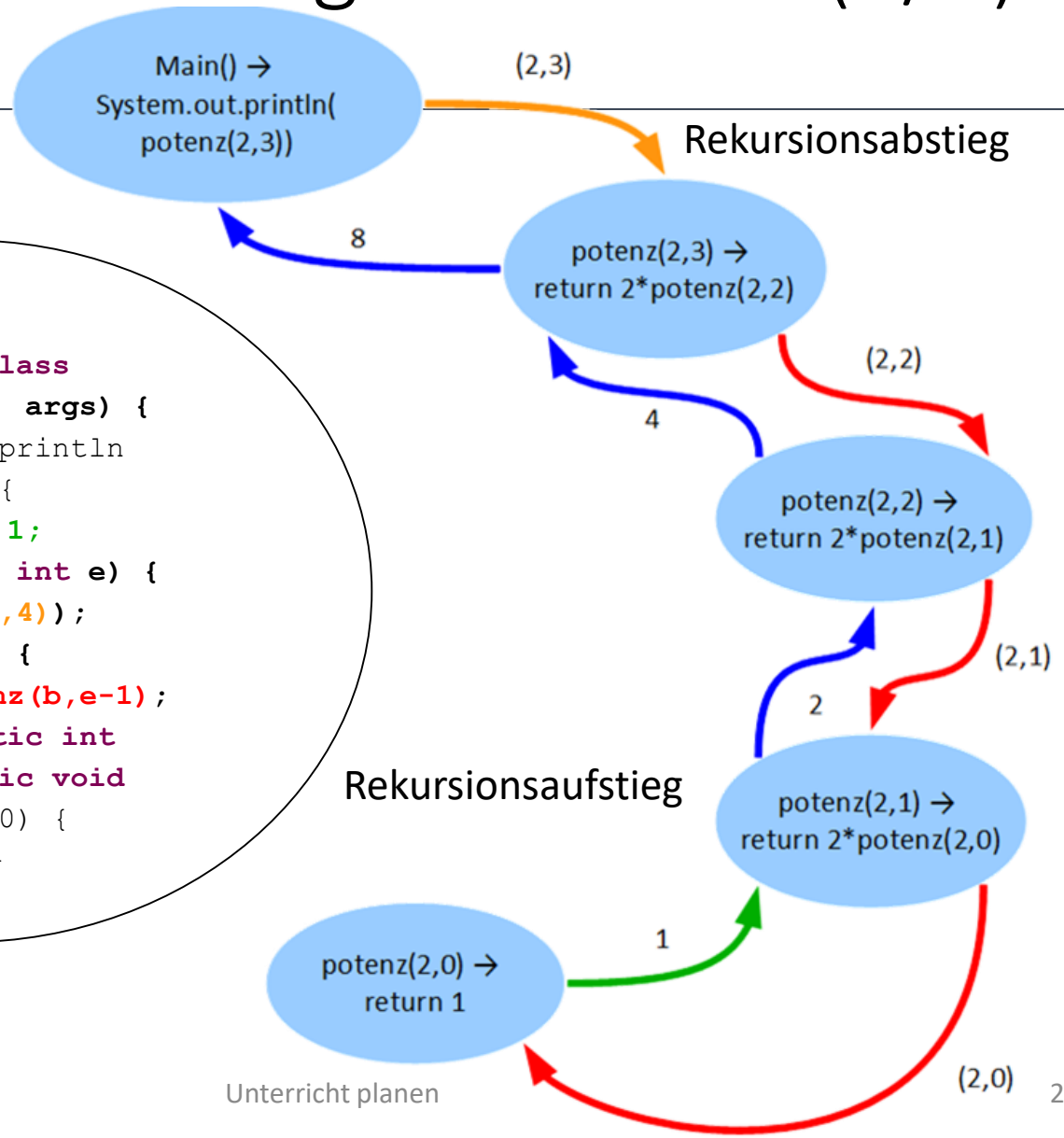
```
public class Endlosrekursion {

   public static void main(String[] args) {
      long zeit = 0;
      zeitVergeht(zeit);
   }

   public static void zeitVergeht(long zeit) {
      System.out.println(zeit);
      zeitVergeht(zeit + 1);
   }
}
```

5.2 Fragen und Aufgaben zur Leseecke

1. Lesen Sie das obige Beispiel zur Rekursion und probieren Sie es aus.
2. Was passiert? Warum?

Algorithmen & Programmieren (4/4)

```
public class
main(String[] args) {
  System.out.println
        else {
      return 1;
potenz(int b, int e) {
    (potenz(2,4));
      Potenz {
return b*potenz(b,e-1);
   public static int
   public static void
      if(e == 0) {
        }}}}}
```



Ausgewählte Infos & Links für den Einstieg in die Rekursion

- Leitprogramm Rekursives Programmieren http://swisseduc.ch/informatik/programmiersprachen/rekursion/docs/rekursion.pdf
- http://www.inf-schule.de/algorithmen/algorithmen/rekursion
- Quicksort mit Lego und Stop Motion
- Sortieralgorithmen mit AlgoRythmics
- Klangsteuerung: I'm sitting in a room

Klangsteuerung I‘m sitting in a room

- I am sitting in a room different from the one you are in now.
- I am recording the sound of my speaking voice and
- I am going to play it back into the room again and again
- until the resonant frequencies of the room reinforce themselves so that any semblance of my speech, with perhaps the exception of rhythm, is destroyed.
- What you will hear, then, are the natural resonant frequencies of the room articulated by speech.
  I regard this activity not so much as a demonstration of a physical fact, but more as a way to smooth out any irregularities my speech might have.“

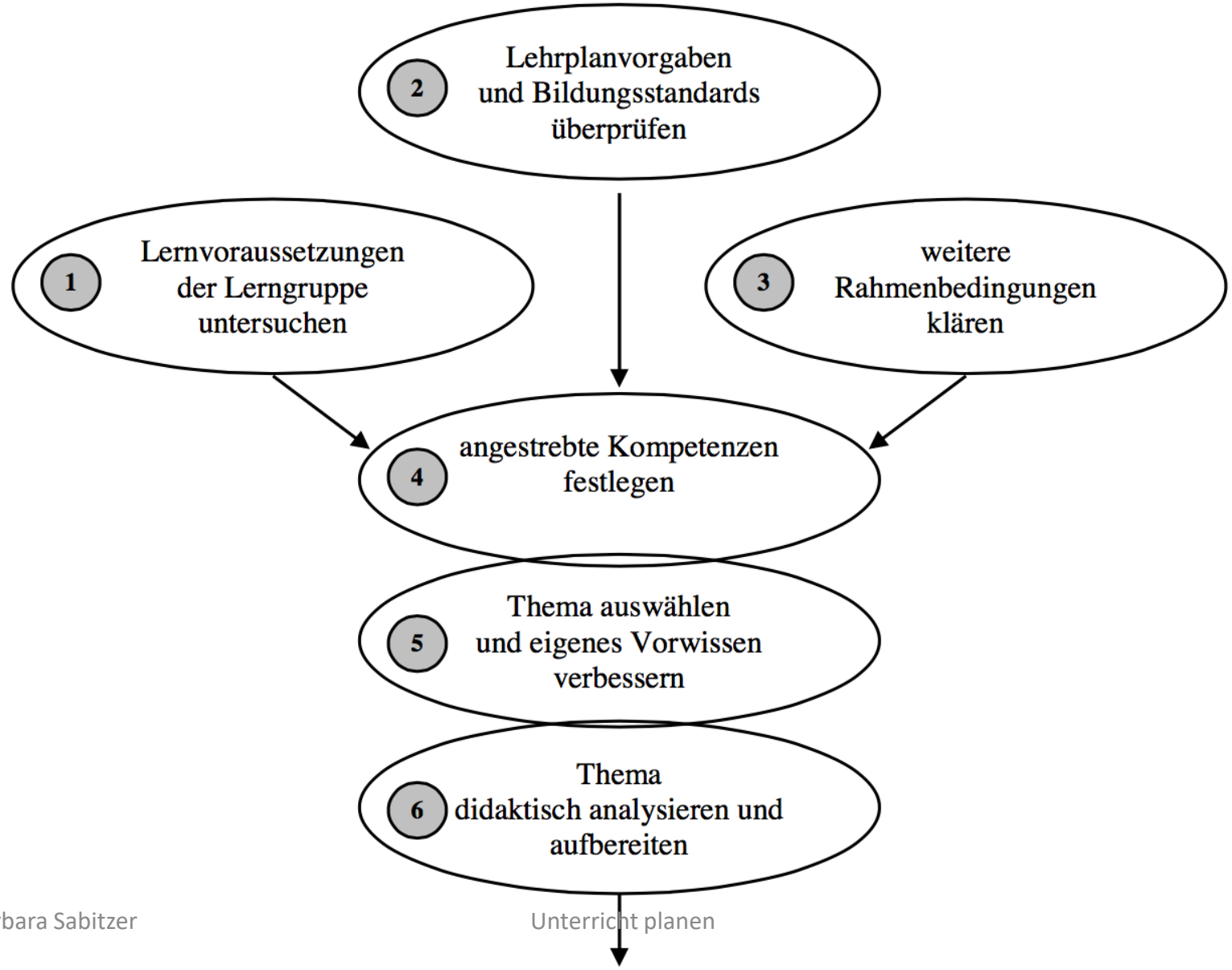
Rekursion

Abschluss – Verständnisfragen an die Studierenden

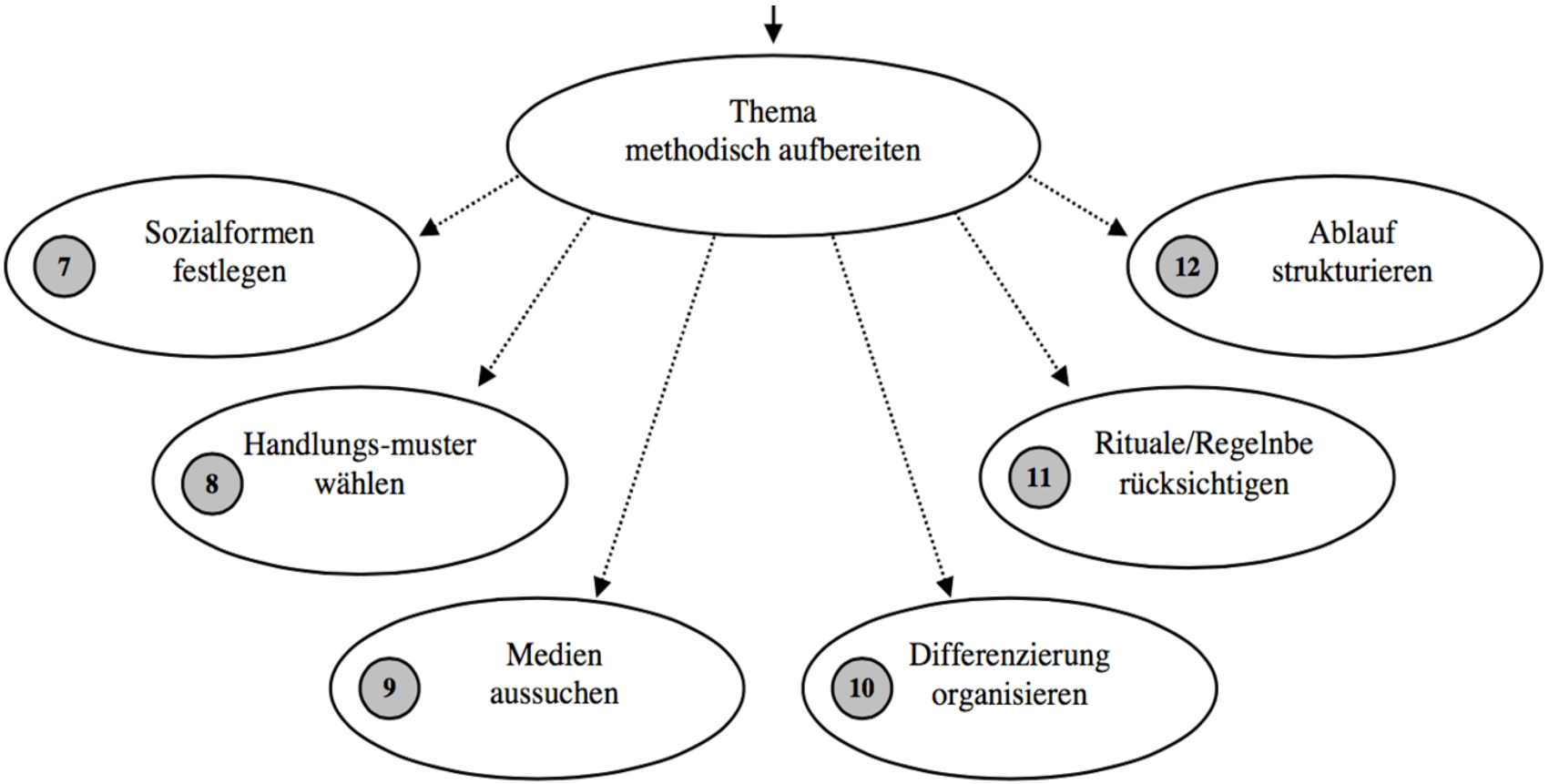
Unterricht planen

Schritte – Grundraster – Kriterien für guten Unterricht

Unterrichtsplanung – 12 Schritte (1/2)



Unterrichtsplanung – 12 Schritte (2/2)



http://www.flensburg-studieren.de/sites/default/files/dateien/service/entwuerfe/Schema%20große%20Unterrichtsvorbereitung.pdf

Grundraster Stundenplanung

1. Vorbereitete Umgebung schaffen
2. Thema der Stunde festlegen (Aufgabenstellung vorformulieren)
3. Bedingungsanalyse
   - Lernvoraussetzungen
   - Lehrvoraussetzungen
   - Richtlinien und Bildungsstandards
4. Didaktische Strukturierung
   - die Ziele festlegen
   - die Inhaltsstruktur klären/Medien vorbereiten
   - den methodischen Gang und die Handlungsstruktur klären
   - die Sozialstruktur anpassen (wer mit wem?)
5. Stundenverlauf planen:
   - Einstieg
   - Erarbeitung
   - Ergebnissicherung
6. Vorüberlegungen zur Auswertung machen

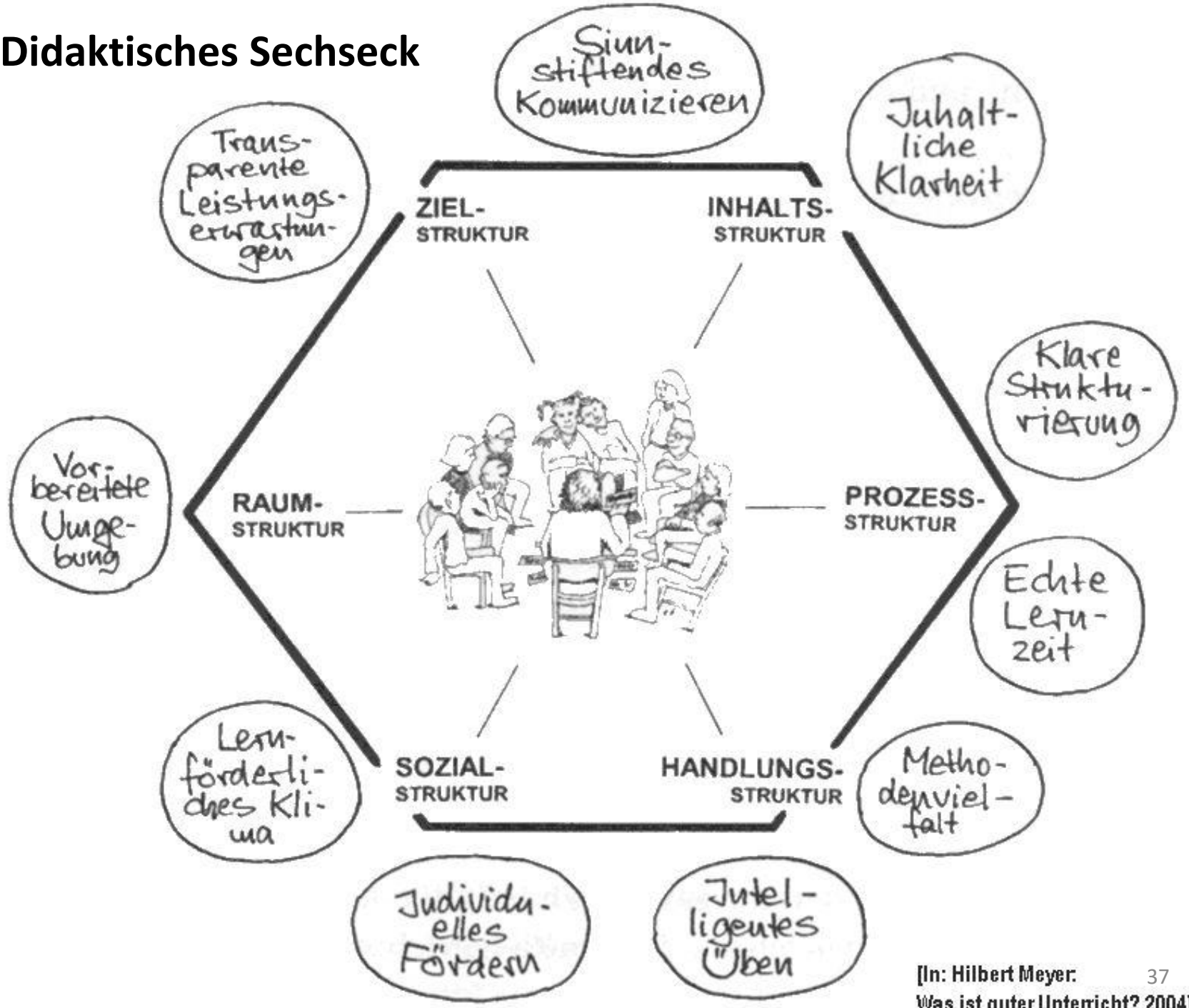
Meyer, Hilbert, *Leitfaden Unterrichtsvorbereitung*. Berlin 2009,4. Auflage, Cornelsen Verlag Scriptor GmbH & Co KG, S.103

3 Grundelemente guten Unterrichts

Erstes Grundelement: **Das relevante Thema**	Zweites Grundelement: **Die konsequente Schülerorientierung**	Drittes Grundelement: **Die konstruktive Atmosphäre**
Transparenz über Inhalt und Rahmen des Unterrichts: Was genau ist Thema des Unterrichts und der Unterrichtsphasen? Was kann gelernt werden? Warum ist das wichtig?	Was wissen die Schüler? Was können die Schüler? Was wollen sie wissen bzw. können?	Lehrer ist „Partner“ und „Chef“ Lehrer ist verantwortlich für: • gegenseitigen Respekt • Regeln • Konsequenz • Klarheit • Rücksichtnahme • Kooperation • Lachen!

Unruh, T., & Petersen, S. (2009). *Guter Unterricht: Praxishandbuch: Handwerkszeug für Unterrichts-Profis (Alle Klassenstufen)*. AOL Verlag.

Didaktisches Sechseck



[In: Hilbert Meyer: Was ist guter Unterricht? 2004]

10 Merkmale guten Unterrichts (1/2)

1. **Klare Strukturierung** des Unterrichts (Prozess-, Ziel- und Inhaltsklarheit; Rollenklarheit, Absprache von Regeln, Ritualen und Freiräumen)
2. **Hoher Anteil** echter Lernzeit (durch gutes Zeitmanagement, Pünktlichkeit; Auslagerung von Organisationskram; Rhythmisierung des Tagesablaufs)
3. **Lernförderliches Klima** (durch gegenseitigen Respekt, verlässlich eingehaltene Regeln, Verantwortungsübernahme, Gerechtigkeit und Fürsorge)
4. **Inhaltliche Klarheit** (durch Verständlichkeit der Aufgabenstellung, Monitoring des Lernverlaufs, Plausibilität des thematischen Gangs, Klarheit und Verbindlichkeit der Ergebnissicherung)
5. **Sinnstiftendes Kommunizieren** (durch Planungsbeteiligung, Gesprächskultur, Schülerkonferenzen, Lerntagebücher und Schülerfeedback)

10 Merkmale guten Unterrichts (2/2)

6. **Methodenvielfalt** (Reichtum an Inszenierungstechniken; Vielfalt der Handlungsmuster; Variabilität der Verlaufsformen und Ausbalancierung der methodischen Großformen)
7. **Individuelles Fördern** (durch Freiräume, Geduld und Zeit; durch innere Differenzierung und Integration; durch individuelle Lernstandsanalysen und abgestimmte Förderpläne; besondere Förderung von Schülern aus Risikogruppen)
8. **Intelligentes Üben** (durch Bewusstmachen von Lernstrategien, Passgenauigkeit der Übungsaufgaben, methodische Variation und Anwendungsbezüge)
9. **Klare Leistungserwartungen** (durch Passung und Transparenz) und klare Rückmeldungen (gerecht und zügig)
10. **Vorbereitete Umgebung** (= verlässliche Ordnung, geschickte Raumregie, Bewegungsmöglichkeiten und Ästhetik der Raumgestaltung) (http://www.staff.uni-oldenburg.de/hilbert.meyer/9290.html)