

Construcción de una teoría didáctica para programación

Federico Gómez

Supervisora: Dra. Ing. Sylvia da Rosa

Grupo de Investigación en Didáctica de la Informática

Instituto de Computación

Facultad de Ingeniería – UdelaR – Uruguay

Agenda

- Motivación
- Antecedentes
- Tema de tesis
- Estudio empírico

Motivación

- Seguir contribuyendo al desarrollo de la disciplina ***Computer Science Education***, específicamente a la ***Didáctica de la Programación***
- Transformar lo que sabemos de ***cómo aprender*** (resultados de investigaciones epistemológicas) en ***cómo enseñar*** (secuencias didácticas para aplicación en aula), proveer fundamentos teóricos para formación docente
- Contribuir al desarrollo de ***PCK – Pedagogical content knowledge***: conjunto de formas de representar la disciplina para hacerla comprensible a estudiantes

Antecedentes

- ***The learning of recursive algorithms and their functional formalization*** (Sylvia da Rosa) [Tesis de doctorado, PEDECIBA Informática, 2005]
- Múltiples investigaciones epistemológicas del área CSE, destacando: Sylvia da Rosa “***Piaget and Computational Thinking***” (2018)
- ***Un Modelo de Investigación en didáctica de la programación*** (Federico Gómez) [Tesis de maestría, PEDECIBA Informática, 2021]

¿Qué sabemos de cómo *aprender*?

Nuestros resultados de investigaciones epistemológicas (**R.I.E** - basados en **Epistemología Genética**) explican proceso de construcción de conocimiento sobre **programación** , destacando tríada ***intra-inter-trans***:

- Etapa ***intra***: conocimiento ***instrumental*** (instancias concretas de algoritmos en la ***acción***)
- Etapa ***inter***: conocimiento ***conceptual*** (lenguaje natural, entender/explicar cómo y por qué funcionan)
- Etapa ***trans***: conocimiento ***formal*** (programas, entender/explicar cómo y por qué funcionan al ejecutarse en ***computadora***)

¿Cómo volcar lo anterior a cómo *enseñar*?

En el **PIMCEU** (2022) incursionamos en cómo responder esta pregunta de la mano de dos teorías didácticas...

Teoría de las variaciones (Marton): aprendizaje surge de ***aspectos críticos*** (propiedades esenciales para comprender un concepto, no evidentes, requieren experimentar sucesivas ***variaciones***)

Teoría de las situaciones didácticas (Brousseau): se basa primero en situaciones ***adidácticas*** (interacciones iniciales, conocimiento desculturizado) y luego en situaciones ***didácticas*** (formulación culturizada)

- ¿Cómo construir una teoría didáctica de ***programación*** integrando nuestros **R.I.E** con las dos teorías?

¿Cómo volcar lo anterior a cómo *enseñar*?

- Entra en escena el enfoque de Hiebert y Stigler para guiar la mencionada construcción:



- Brinda lineamientos **genéricos** instanciables luego a cualquier dominio **específico**
- ***Sustained learning opportunities***: sistemas temporalmente estables que emergen durante las clases para crear contextos en los que se produce aprendizaje

Tema de tesis



- 1ra instanciación del **sistema S1** para **programación** (dar significado a las dos cajas y flecha) en base a:
 - Integración de nuestros **R.I.E** con teoría de las **variaciones**, teoría de las **situaciones didácticas**
 - Experiencia docente y en investigación
 - Estudio empírico para instanciación del sistema **S1**
 - Análisis de resultados y primera caracterización de SLO y la teoría didáctica (en proceso)

Estudio empírico

- Diseño, aplicación (y reflexión) de una secuencia didáctica para introducir un tema específico en un curso inicial de programación imperativa
- Tema elegido: *arreglo con tope* con cuatro operaciones básicas: *inicialización*, *inserción*, *listado* y *búsqueda* (nexo entre estructuras estáticas y dinámicas, complejidad adecuada para trabajar ideas fundamentales)
- Secuencia didáctica compuesta por cuatro grupos de actividades (A, B, C, D) que fomenten el aprendizaje de la estructura y las cuatro operaciones durante su aplicación en clase (**SLO**). Aplicada en **dos** grupos de estudiantes

Estudio empírico

Grupo A: Manipulación de una instancia concreta



- Situación **adidáctica**: Estantería de discos, se colocan de a uno por vez de izquierda a derecha, ¿cómo saber cuántos hay sin contarlos? (**struggle-first**). Necesidad de “papel” (futuro **tope**, sienta base como **aspecto crítico**)
- Etapa **intra**: Inicializar “papel” e insertar a mano un disco
- Etapa **inter**: pseudocódigo para **inicialización, inserción**

Estudio empírico

Grupo A: Manipulación de una instancia concreta



E: Estantería, nuevo disco

S: Disco colocado en el estante

M: SI hay menos de 50 discos ENTONCES

sumo 1 al papel que dice cuántos discos hay
coloco nuevo disco enseguida del último

FIN

Estudio empírico

Grupo B: Formalización en lenguaje de programación

- Inicio pasaje a etapa **trans**: Correspondencia (discos → celdas) y (“papel” → tope)
- Proponer estructura de datos para representación en **lenguaje de programación** (en base a conocimiento previo de arreglos y registros)
- Se introduce terminología **arreglo con tope** (culturizado)
- Situación **didáctica**: pequeños ejercicios; practicar sintaxis y manejo, similitudes y diferencias con arreglo clásico (fomentar observación de variaciones para discernir importancia del **tope** como **aspecto crítico**)

Estudio empírico

Grupo B: Formalización en lenguaje de programación

(* formulación inicial *)

```
const N = 50;
```

```
type Estanteria = record
```

```
    discos: array[1..N] of integer;
```

```
    papel: 0..N;
```

```
end;
```

(* formulación culturizada *)

```
const N = 50;
```

```
type ArregloTope = record
```

```
    arre: array[1..N] of integer;
```

```
    tope: 0..N;
```

```
end;
```

Estudio empírico

Grupo C: Operaciones de inicialización e inserción

- Escribir, compilar y ejecutar subprogramas para ambas
- Transformar **pseudocódigo** de inicialización e inserción en **subprogramas** (profundizar pasaje de **inter** a **trans**)
- Operaciones nuevas, se prevén errores de **sintaxis** (combinación de arreglos con registros por primera vez) y de **ejecución** (ej: salida de rango, celdas con “basura”), explicado por **ley extendida de la cognición** (R.I.E)
- **Automatización** (R.I.E), mecanismo que ayuda a detectar errores, ejecución manual sobre arreglo en papel

Estudio empírico

Grupo C: Operaciones de inicialización e inserción

```
procedure inicializar (var a: ArregloTope);  
begin  
    a.tope := 0  
end;  
  
procedure insertar (var a: ArregloTope; d: integer);  
begin  
    if act.tope < N then  (* si hay menos de 50 discos *)  
    begin  
        a.tope := a.tope + 1;  (* sumo 1 al papel *)  
        a.arre[a.tope] := d    (* coloco nuevo disco *)  
    end  
end;
```


Estudio empírico

Grupo D: Operaciones de listado y búsqueda

- Escribir, compilar y ejecutar subprogramas para ambas
- Se apoyan en conocimiento previo sobre versiones previas en arreglo clásico (razonables similitudes y diferencias).
Herramienta cognitiva **generalización** (R.I.E)
- Profundización en experimentación de **variaciones**, conocimiento nuevo directamente en etapa **trans**
- Necesidad de variar límite superior de recorrida (deja de ser la constante que define el **tamaño** y pasa a ser el **tope**, profundiza su carácter de **aspecto crítico**)

Estudio empírico

Grupo C: Operaciones de listado y búsqueda

```
procedure listado (a: ArregloTope);
var i: integer;
begin
    for i := 1 to act.tope do
        writeln (a.arre[i])
    end;

function busqueda (a: ArregloTope; d: integer): boolean;
var i: integer;
begin
    i := 1;
    while (i <= a.tope) and (a.arre[i] <> d) do
        i := i + 1;
    busqueda := i <= a.tope
end;
```

Estudio empírico: Análisis preliminar

- Ajustes y mejoras (tras aplicación en primer grupo) tuvieron impacto positivo en generación de SLO en el segundo grupo
- Evidencia recolectada (hojas de papel y archivos fuente con respuestas de estudiantes, observaciones realizadas) así lo sugiere (análisis en profundidad en proceso)
- En ambos grupos, el carácter gradual y progresivo permitió completar con éxito las actividades. No obstante, en el primer grupo se observaron más dificultades (presumiblemente por el planteo de las actividades)

¡Muchas gracias!

¿Preguntas?