

Stratégies explicites

Didactique de l'informatique - expérience de recherche

Olivier Goletti, UCLouvain, INGI, olivier.goletti@uclouvain.be,  [@OGoletti](https://twitter.com/OGoletti)
Prof. Kim Mens, UCLouvain, INGI, kim.mens@uclouvain.be,  [@kim_mens](https://twitter.com/kim_mens)
18/05/22 - #Didapro22



Contexte

**Promoting Learning Transfer in Computer
Science Education by Training Tutors
to use Explicit Programming Strategies**

**Promouvoir le Transfert des Apprentissages dans
l'enseignement de l'informatique en formant les éducateurs
à l'utilisation de Stratégies Explicites pour programmer**

Besoin de **stratégies explicites** pour **programmer**

- "human-executable procedure for accomplishing a programming task"
- **Basées sur la littérature**
- Pour réduire **charge cognitive** des étudiants
- Favoriser le **transfert**
- Plus **d'étayage** le cadre de méthodologies d'enseignement moins guidées
- Appliquées par des **tuteurs**

Deux stratégies en particulier

Explicit Tracing

Method Name:	<i>wildMystery</i>	Method Name:	<i>wildMystery</i>
Name	Value	Name	Value
<i>n</i>	<i>4.7</i>	<i>n</i>	<i>31.32</i>
<i>x</i>	<i>x 2</i>	<i>x</i>	<i>x 3</i>
<i>y</i>	<i>1.8</i>	<i>y</i>	<i>x 8 7 7</i>
<i>output</i>	<i>- 2 - 8</i>	<i>output</i>	<i>+ 3 2 - 3 - 7 7</i>
Return		Return	

Figure 1: Two memory tables for 2 calls to the same method

Subgoal-labeled Worked examples

Figure 1. Subgoals identified through TAPS protocol.

Subgoals for evaluating and writing expression (assignment) statements	
A. Evaluate expression statement	B. Write expression statement
1. Determine whether data type of expression is compatible with data type of variable 2. Update variable for pre based on side effect 3. Solve arithmetic equation 4. Check data type of copied value against data type of variable 5. Update variable for post based on side effect	1. Determine expression that will yield variable 2. Determine data type and name of variable and data type of expression 3. Determine arithmetic equation with operators 4. Determine expression components 5. Operators and operands must be compatible
Subgoals for evaluating and writing selection statements	
A. Evaluate selection statement	B. Write selection statement
1. Diagram which statements go together 2. For if statement, determine whether expression is true or false 3. If true – follow true branch, if false –follow else branch or do nothing if no else branch	1. Define how many mutually exclusive paths are needed 2. Order from most restrictive/selective group to least restrictive 3. Write if statement with Boolean expression 4. Follow with true bracket including action 5. Follow with else bracket 6. Repeat until all groups and actions are accounted for
Subgoals for evaluating and writing loops.	
A. Evaluate loops	B. Write loops
1. Identify loop parts 2. Determine start condition	1. Determine purpose of loop 2. Pick a loop structure (while, for, do, while)

Comment ?

- Expliquer et nommer la stratégie
 - Exemplifier la stratégie
 - Être cohérent et systématique
 - Encourager les étudiants à réutiliser la stratégie
-
- Au feeling
 - À adapter selon votre ressenti

Explicit Tracing

Explicit tracing

- Stratégie d'exécution de code
- par écrit
- de façon systématique
- pour soulager la charge mentale
- représentation de la mémoire standardisée



Un exemple

```
def count_a(s):  
    cnt = 0  
    for c in s:  
        if c == "A":  
            cnt += 1  
    return cnt  
  
result = count_a("BANANA")
```

Quand ?

- lors de l'introduction d'un nouveau concept
- en cas de bug
- en cas de surcharge cognitive
- en cas de conflit entre ce que l'étudiant prédit et ce que le code donne

Passons dans le document

Subgoal-labeled Worked Examples

Subgoal-labeled Worked Examples

- Worked examples
 - Exemples contextuels résolus
 - étapes de résolution
 - Pour généraliser
 - exemples variés (voire contre exemples)
- Subgoal-labels
 - labels génériques
 - Mise en évidence de la structure

Exemple 1 - lecture de condition

1. Rassembler les expressions qui vont ensemble
2. Evaluer la condition
3. Tracer la branche sélectionnée :
 - Si True, suivre la branche vraie,
 - Si False, suivre les branche elif suivante ou else si il y en a une. Ne rien faire sinon

Exemple 2 - écriture boucle

1. Choisir le structure de boucle appropriée (while, for)
2. Définir et initialiser les variables
3. Déterminez la condition d'arrêt (ou de continuation)
4. Écrire le corps de la boucle (mettre à jour la variable de contrôle jusqu'à la condition d'arrêt si nécessaire)

Quand ?

- lors de l'introduction d'un nouveau concept
- presque systématiquement au début (vocabulaire commun)
- en cas de difficulté (ou d'erreur)
- en cas d'oubli
- en cas de surcharge cognitive

Passons dans le document

Attention !

- Préparer à l'avance si possible
- Rédiger un exemple est non trivial
- Trouver les bons labels est non trivial
- Fonctions/classes/structures de données à suivre.