

הוראת הפשטה בקורסי מבוא

Teaching Abstraction in Introductory Courses

Herman Koppelman

המאמר השלם ניתן להורדה בכתובת:

<http://db.grinnell.edu/sigcse/iticse2010/Program/viewAcceptedProposal.asp?sessionType=paper&sessionNumber=43>

הוצג בכנס ITiCSE שהתקיים באנקרה טורקיה 2010

המאמר עוסק בבעיות של לומדים מתחילים כאשר הם נתקלים לראשונה בהפשטה, דרך ניתוח של 3 משימות מקורסי מבוא, במטרה להבין למה הפשטה קשה למתחילים.

הקדמה

הפשטה היא נושא מרכזי במדעי המחשב. בדו"ח של וועדת תכנית הלימודים 2008 מצוין שצריך להדגיש הפשטה לכל אורך תכנית הלימודים ושכל בוגר מדעי המחשב צריך להכיר בחשיבות ההפשטה. אבל מצד שני, מחקרים הראו שקשה להגיע לשליטה בנושא הזה ויש אפילו שהעלו ספקות האם ניתן בכלל ללמד את הנושא.

המאמר מתמקד בשתי שאלות: באילו קשיים נתקלים לומדים מתחילים כשהם לומדים הפשטה? באילו דרכים ניתן לסייע ללומדים?

המאמר מנתח 3 משימות תכנותיות שהשימוש בהפשטה יכול לסייע לפתירתן ומנתח את הכוחות הגורמים קשיים למתחילים ומקשים עליהם להשתמש בהפשטה. המשימות לקוחות משלושה תחומים שונים: לולאה מקוננת, שאילתה על מסד נתונים רלציוני, ופונקציה רקורסיבית.

דוגמה 1 - לולאה מקוננת

נתון מערך של מספרים שלמים $a[1..100]$

המשימה: בדוק האם קיים איבר $a[i]$ במערך שמקיים את התכונה שסכום ספרותיו שווה ל-10.

(ראה פתרון אפשרי למשימה בעמוד הבא.)

מאפיין מכריע של האלגוריתם הזה הוא לולאות מקוננות אבל לולאות `while` מקוננות יוצרות הרבה שגיאות. למשל, במחקר של מילר, גינת והברמן יש ניתוח מעניין שדומה למקרה הזה. במחקר שלהם התברר שמתחילים התקשו לזהות את הצורך בלולאה מקוננת. ואם כבר הצליחו לזהות, היתה להם בעיה להפריד בין התנאים ששלטו על שתי הלולאות.

לאלגוריתם הנוכחי (שמופיע בעמוד הבא), יש מספר רמות הפשטה אבל רק שתיים מהן רלבנטיות בהקשר הזה. הרמה הראשונה שייכת למערך ולתהליך סריקתו במטרה למצוא איבר שמקיים תכונה מסוימת (שלא נפרט אותה ברמה הזו). ברמה השנייה, נפרט את התכונה ונכתוב פרוצדורה שתבדוק האם מספר שלם מסוים מקיים את התכונה. אבל שתי רמות ההפשטה האלה מעורבות וכאשר קוראים את האלגוריתם, הקורא צריך לקפוץ הלוך ושוב בין שתי הרמות. כלומר, מתחילים ברמה של המערך ($i:=1$), עוברים לרמה שנייה (מתחיל עם $sum:=0$ ומסתיים בקידום המונה $i:=i+1$) וכן הלאה.

פתרון אפשרי למשימה יראה כך :

```
i := 1;
found := FALSE;
WHILE (i < 101) AND NOT found DO
    number := a[i];
    sum := 0;
    WHILE number > 9 AND sum < 10 DO
        lastDigit := number MOD 10;
        number := number DIV 10;
        sum := sum + lastDigit;
        found := sum = 10;
    END;
    i := i + 1;
END;
```

ניתן להפריד באלגנטיות את שתי רמות ההפשטה אם משתמשים בפונקציה Satisfies ככלי להפשטה :

```
I := 1;
found := FALSE;
WHILE (I < 101) AND NOT found DO
    found := Satisfies (a[i]);
    I := I + 1;
END;
```

האלגוריתם השני זהה אבל השימוש בפונקציה מאפשר הפרדה לשתי רמות הפשטה ברורות ונפרדות. כתוצאה מכך, המתכנת יכול להתרכז ברמה אחת בכל פעם. משמעות ההפשטה היא שלמתכנת לא צריך להיות אכפת מהדרך בה הפונקציה בודקת את קיום התכונה (הוא יכול להתעלם מהבדיקה). בדרך זו, תלמיד לא צריך לטפל בו זמנית ברמות הפשטה שונות.

מתחילים לא חושבים על פתרון כזה באופן ספונטני וזה לא מפתיע. בואו נשווה איך בני אדם יתמודדו עם בעיה כזו. תיאור לפתרון "אנושי ידני" יראה כך :

- בדוק את האיבר הראשון במערך, סכם את ספרותיו ובדוק האם הסכום הזה שווה ל- 10
- אם כן – עצור
- אחרת, המשך באופן דומה לאיברים הבאים.

בפתרון "הידני" הזה רמות ההפשטה מעורבות וקידוד של הפתרון הזה לשפת תכנות יוביל ללולאות מקוננות. לכן, אין סיבה טובה לכך שהפשטה תהיה טבעית למתחילים במקרה כזה. שימוש בהפשטה דורש חשיבה על הבעיה בצורה שונה לגמרי ולכן צריך ללמד את השימוש בהפשטה בצורה מפורשת וברורה.

שימוש במשימה כזו יכול לאפשר הזדמנות טובה להציג ולהדגים את עקרון ההפשטה. אפשר להראות לתלמידים שזה מועיל להשתמש כאן בהפשטה מנקודת מבט פרקטית. הסתרת הלולאה המקוננת הופכת את הפתרון לברור יותר ומפחיתה את הסכנה בשגיאות. Gries אפילו ממליץ "להסתיר תמיד לולאות מקוננות" באופן דומה. לרוע המזל, בספרי לימוד לא מציינים זאת ומשתמשים בלולאות מקוננות מבלי להזכיר הפשטה.

דוגמה 2 – שאילתה

נתונות שתי טבלאות של מסד נתונים רלציוני (יחסי) – טיסות ומטוסים.

Flight

flight_nr	departure	destination	plane_nr
265	Rome	Athens	13
415	Berlin	Paris	28
598	Oslo	Sofia	17

Plane

plane_nr	capacity	company	type
13	235	Air Berlin	Airbus 310
17	210	Ibria	Airbus 310
28	240	Alitalia	Boeing 747

המשימה היא לכתוב שאילתת SQL שבודקת את הפרטים של טיסות המבוצעות על ידי מטוס Airbus 310 (הפרטים הם מספר טיסה, מקום המראה, מקום נחיתה). פתרונות אפשריים למשימה (עם ובלי הפשטה) אפשר למצוא במאמר.

דוגמה 3 – פונקציה רקורסיבית

נתבונן בפונקציה NumberOfDigits שסופרת כמה ספרות יש במספר שלם חיובי.

NumberOfDigits (posint n): posint

IF n < 10 THEN

 RETURN 1

ELSE

 RETURN NumberOfDigits (n div 10) + 1

ניתוח של הפונקציה מבחינת הפשטה ניתן למצוא במאמר.

דיון

הדוגמאות המובאות במאמר מראות שהשימוש בהפשטה לא קל למתחילים. ספרי הלימוד מעודדים לפתרון "ידני" אבל פתרון כזה לא עוזר למתחילים כי יש פער בין הפתרון ה"ידני" לבין השימוש בהפשטה.

באופן כללי, יש פער בין דרך ההוראה בקורסי המבוא לבין רכישת המיומנויות של הפשטה. בקורסי המבוא יש דגש על הבנת הזרימה של התכנית (בעזרת טבלאות מעקב למשל). אבל זה לא מספיק. לומדים צריכים ללמוד גם איך לזהות סוגים שונים של הפשטה בפיסת קוד, ללמוד איך להתייחס לקריאה של פונקציה לא רק כדרך להעברת בקרה למקום אחר אלא גם כמכניזם להדחקת מידע לא רלבנטי. הם צריכים להתמקד בהשפעה של הקריאה לפונקציה ולהתעלם מהנטייה לבדוק את יישומה.

שימוש בהפשטה מחייב הסתכלות חדשה על הבעיה ולכן, מתחילים לא יבינו הפשטה באופן ספונטני. הם צריכים לעבור "כאבי למידה" כדי ללמוד את המושג.

מחבר המאמר ממליץ להתחיל את הוראת ההפשטה בשלב מוקדם ככל האפשר. הדוגמאות במאמר ממחישות שמושג ההפשטה ניתן להצגה גם עם דוגמאות פשוטות. הפשטה היא לא מושג שניתן לשלוט בו באופן שלם. סטודנטים לומדים אותו בהדרגה דרך דוגמאות קונקרטיות ולכן נדרש זמן לפני שמבינים את הנושא כהלכה.

ההמלצה הבאה היא ללמד הפשטה באופן מודע. מומחים במדעי המחשב משתמשים בסוגים שונים של הפשטה הרבה פעמים בלי להיות מודעים לכך. הוראה מודעת פירושה שהמורה צריך להדגיש איפה היה שימוש בהפשטה ולקרוא לכך בשם. מורים צריכים גם להראות מדוע הפשטה מסוימת בדוגמה ספציפית היא חדשה ללומדים.

באופן כללי יותר, הוראת תכנות היא לא רק הוראה של מאפייני שפה, אלא גם עקרונות ואסטרטגיות שצריך ליישם אותם בפיתוח תוכנית. לפי Gries תכנות הוא יותר מסתם אוסף של עובדות. זוהי מיומנות, והוראת תכנות כמיומנות משמעותה יותר מהוראת עובדות.

המלצה נוספת של מחבר המאמר היא להדגיש את היתרונות של שימוש בהפשטה. להרבה סטודנטים יש קונוטציה של משהו קשה כשהם שומעים על הפשטה. מחקרים הראו שסטודנטים מאפיינים הפשטה כמשהו מורכב, דורש חשיבה מעמיקה, וקשה להבנה. כדי להתמודד עם התפיסה הזו, מורים צריכים להראות שהפשטה מקלה על החיים, ברגע שאתה שולט בעקרון. מורים יכולים להשוות פתרונות (עם ובלי הפשטה), ויכולים להראות יתרונות של שימוש בהפשטה.