

דמוקרטיה בכיתה: תרגיל לפתיחת הקורס הראשון במדעי המחשב

Democracy in the Classroom: An Exercise for the First Days of CS1

Don Blaheta

המאמר השלם ניתן להורדה בכתובת:

<http://db.grinnell.edu/sigcse/iticse2009/Program/viewAcceptedProposal.asp?sessionType=paper&sessionNumber=181>

הוצג בכנס ITiCSE שהתקיים בפריז צרפת 2009

בחירת הדוגמה הראשונה לשימוש בקורס היא חשובה במיוחד. המאמר מציע משימה של ספירת קולות כדוגמה טובה לפתיחת הקורס הראשון במדעי המחשב. המשימה כוללת טיפול באלגוריתמים, ניהול נתונים, עיבוד מקבילי ורעיונות מרכזיים נוספים.

איך להתחיל את השיעור הראשון בקורס הראשון במדעי המחשב? יש כאלה שאוהבים להתחיל עם התפתחות המטאפורות של הגישה מונחית העצמים (דרך משחק תפקידים למשל. ראה "משחק תפקידים להכרות עם תכנות מונחה עצמים" שפורסם בגליון ינואר 2008 של "הבטים בהוראת מדעי המחשב"). יש כאלה שמעדיפים להתחיל עם דגש על חשיבה אלגוריתמית (למשל בניית אלגוריתם ליצירת כריך של חמאת בוטנים וריבה).

המאמר מציע משימה נוספת שמומלץ להתחיל איתה את הקורס: **ספירת קולות**. תיאור המשימה: הסטודנטים צריכים לבחור את שעת הקבלה של המרצה מבין מספר אפשרויות נתונות (בסה"כ יבחרו 3 משבצות זמן). הואיל ויבחרו יותר מאפשרות (משבצת זמן) אחת, הסטודנטים יצביעו על ידי דרוג השעות המועדפות עליהם. יתרונות המשימה: שימוש במושג המוכר לסטודנטים (הצבעה בבחירות) אבל רק מעט מהסטודנטים אכן חשבו עליו לעומק. בפרט, מערכות הצבעה המבוססות על העדפה לפי דירוג הן מערכות שקל להשתמש בהן אבל חישוב התוצאות דורש חשיבה יותר מעמיקה. כמו כן, המשימה בהירה ונגישה לכל הסטודנטים וחושפת מספר הבטים שונים של תהליך התכנות.

ההוראות לסטודנטים: (תיאור המשימה חולק לסטודנטים לפני השיעור הראשון)

דרג את השעות המועדפות עבורך לשעת קבלה (מפגש עם מרצה הקורס). כאשר "1" מצייין את הבחירה הטובה ביותר עבורך. אם אינך יכול להגיע בשעה מסוימת (מבין האפשרויות המוצגות בהמשך) – השאר אותה ריקה (השאר את המשבצת ריקה).

להלן 6 אפשרויות לבחירה: (לפחות אפשרות אחת מתנגשת עם שעת ההרצאה כדי ליצור בכוונה קונפליקט)

יום ראשון 10:00-11:00

יום שני 13:00-14:00

יום שלישי 8:00-9:00

יום רביעי 9:00-10:00

יום רביעי 15:00-16:00

יום חמישי 10:00-11:00

לאחר שהסטודנטים דרגו את השעות המועדפות עליהם, הטפסים נאספו והתוצאות חושבו על ידי סגל ההוראה בקורס.

בשיעור השני הסטודנטים התבקשו לחשוב איך ניתן להשתמש בתוצאות ההצבעה כדי להחליט אילו 3 משבצות זמן (מתוך 6) ייקבעו כשעות קבלה של מרצה הקורס. השלב הבא היה "להפיק" אלגוריתמים שהתבססו על הצעות הסטודנטים תוך הדגשת הבעיות שיכולות להתעורר מבחינת יעילות או איכות הפתרון, ותוך סיוע לסטודנטים כדי שיתקנו ויעדנו את האלגוריתמים שלהם. הדיון התחיל עם פיתוח של פרוצדורה לביצוע ספירת הקולות. אבל הואיל והפורמט הנוכחי של הנתונים אינו נוח לשימוש (מספר גדול של טפסים שעל כל אחד מהם רשומות ההעדפות של סטודנט אחד), הדיון התחיל בעצם בדרכים שונות להבניית הנתונים ואיך לרכז בתוך המבנים האלה את הנתונים עצמם.

שני רעיונות נוספים שניתן לדון בהם בשלב הזה (בנפרד או יחד עם הנ"ל) הם איך להאיץ את התהליך על ידי ביצוע של חלק מהפעולות במקביל וכן איך לטפל (להתייחס) לנתונים "רועשים" (noisy data) שלא מתאימים למסגרת שהחליטו עליה.

תכנון בסיסי של האלגוריתם

הגרסה הראשונה של האלגוריתם (לספירת קולות) צפויה להראות כך: בחר משבצת זמן עם המספר הגדול ביותר של הצבעות "1".

לגרסה הזו יש יתרון – היא פשוטה ומספקת הזדמנויות לבקש מהסטודנטים עידון נוסף (למשל, לשאול: איך נדע לאיזו משבצת זמן יש את מספר ה- "1" הגדול ביותר?) עד שיגיעו לגרסה שנראית דומה לאלגוריתם ממשי.

אחרי שהאלגוריתם היה מוגדר פחות או יותר בצורה ברורה, הריצו אותו ובדקו את תוצאות הריצה.

עידון האלגוריתם

האלגוריתם הפשוט הנ"ל יכול ליצור מספר בעיות. למשל, האלגוריתם לא מתחשב בכלל בכך שיש משבצות זמן שלא מתאימות לחלק מהסטודנטים (גם אם רבים אחרים בחרו דווקא אותן). בעיה זו מובילה לדיון על מערכות הצבעה המבוססות על העדפה לפי דירוג. להלן מספר שיטות שקל לחשב אותן ידנית (תוך שימוש במערך דו ממדי) ושקל לייצר איתן לפחות 2-3 תוצאות "טובות":

- בחר את 3 משבצות הזמן עם הכי מעט קונפליקטים
- בחר את 3 משבצות הזמן עם הכי הרבה "1"-ים
- בחר את 3 משבצות הזמן עם הכי הרבה "1"-ים או "2"-ים
- בחר את 3 משבצות הזמן עם הכי הרבה "1"-ים או "2"-ים או "3"-ים
- בחר את 3 משבצות הזמן עם הציון הנמוך ביותר כאשר מחשבים את ממוצע ההצבעות למשבצת זמן:
 - בהתעלם מרווחים
 - כאשר מחשבים רווחים כ- 7 (גבוה באחד ממספר המשבצות המבוקשות כי מחפשים 6 משבצות זמן)
 - כאשר מחשבים רווחים כ- 10 או 20 (גבוה בהרבה ממספר משבצות הזמן)

הדיון חשוב כדי להדגים לסטודנטים את החשיבות של בחירת האלגוריתם.

כך למשל, היו תנאים שהסטודנטים לא הצליחו לפתור אותם. כמו, איך לתת משקל גדול יותר לבחירה שנייה (או שלישית) של סטודנטים כאשר הבחירה הראשונה שלהם לא עלתה יפה. או איך להבטיח שלכל סטודנט תינתן לפחות משבצת זמן אחת שתתאים לו. הדיון הני"ל על האלגוריתם יכול להמחיש גם את החשיבות של מבני נתונים (קשה לעבוד כל הזמן עם הטפסים ולכן כתבו על הלוח בתוך מערך רק את התוצאות המשוקללות. תוספת המקום על הלוח חסכה זמן ביצוע).

יעילות

ביצוע האלגוריתם הפשוט ביותר (בחר משבצת זמן עם המספר הגדול ביותר של הצבעות "1") בוצע ע"י סיכום ה-"1"-ים לתוך מערך דו ממדי שהיה על הלוח. אבל זה אלגוריתם לא כל כך יעיל ויכול לעודד דיון על חישוב מקבילי (העבודה תוכל להיות יעילה יותר אם נחלק אותה בין מספר אנשים).

נתונים רועשים

למרות שההנחיות למילוי הטפסים היו פשוטות, היו סטודנטים שלא ביצעו אותן במדויק. למשל כתבו 0 (אפס) במקום להשאיר רווח או כתבו "-" (מקף) במקום רווח, או כאלה שסימנו v במקום מספרים. והיו כאלה שנתנו אותו ציון לשתי משבצות זמן שונות – כל אלה הובילו לדיון חשוב נוסף (איך מטפלים בנתונים כאלה?).

סיכום

הכי חשוב – התוצאה של החישוב הייתה חשובה לסטודנטים כי הפעילות הזו באמת קבעה את השעות בהן יוכלו להגיע לשיחה עם המרצה (התוצאה משמעותית עבורם). היו להם הצעות מגוונות ויצירתיות איך לפתור את הבעיות בזכות האכפתיות שלהם מהתוצאות וזה לא היה קורה אם המשימה הייתה מופשטת. המשימה מאפשרת להתחיל את הקורס עם דגש על חשיבה. היא ממחישה לסטודנטים שאלגוריתם הוא לא משהו תיאורטי ושיש לו השלכות יישומיות לחייהם. גם אם לא יכולים לסיים את המשימה עם תכנית מלאה (יידרשו מספר חודשים עד שיהיה להם כל הידע) – עדיין יש להם לפחות מושג ראשוני וכיוון כללי לאן הקורס עומד להתפתח.