חידה לחימום

נתונים שני מספרים טבעיים M ו-N (כך ש-$M < N$) בעלי אותה הזוגיות (שניהם זוגיים או שניהם אי-זוגיים).

המספרים הטבעיים מ-M עד N מסודרים בשורה, ושני שחקנים משחקים במשחק.

כל אחד מהם, בתורו, בוחר את אחד המספרים שנותרו בשורה – ומסיר אותו.

המשחק נמשך עד שלבסוף נותרים שני מספרים בלבד.

אם הם זרים זה לזה – השחקן הראשון מנצח. אם יש להם מחלק משותף גדול מ-1 – השחקן השני מנצח.

כתבו תכנית מחשב, המקבלת כקלט את M ו-N, מחליטה האם ברצונה להיות השחקן הפותח או השחקן השני, ותשחק כך שהיא תנצח תמיד.

דוגמא למשחק: הקלט הוא $M = 7, N = 11$. התכנית תבחר להיות השחקן הפותח, ותוריד את המספר 9. המשתמש יוריד, למשל, את 8. התכנית תוריד את 11, ותותיר את 7 ו-10, הזרים זה לזה, ותנצח.

מבני נתונים ויעילות אלגוריתמים

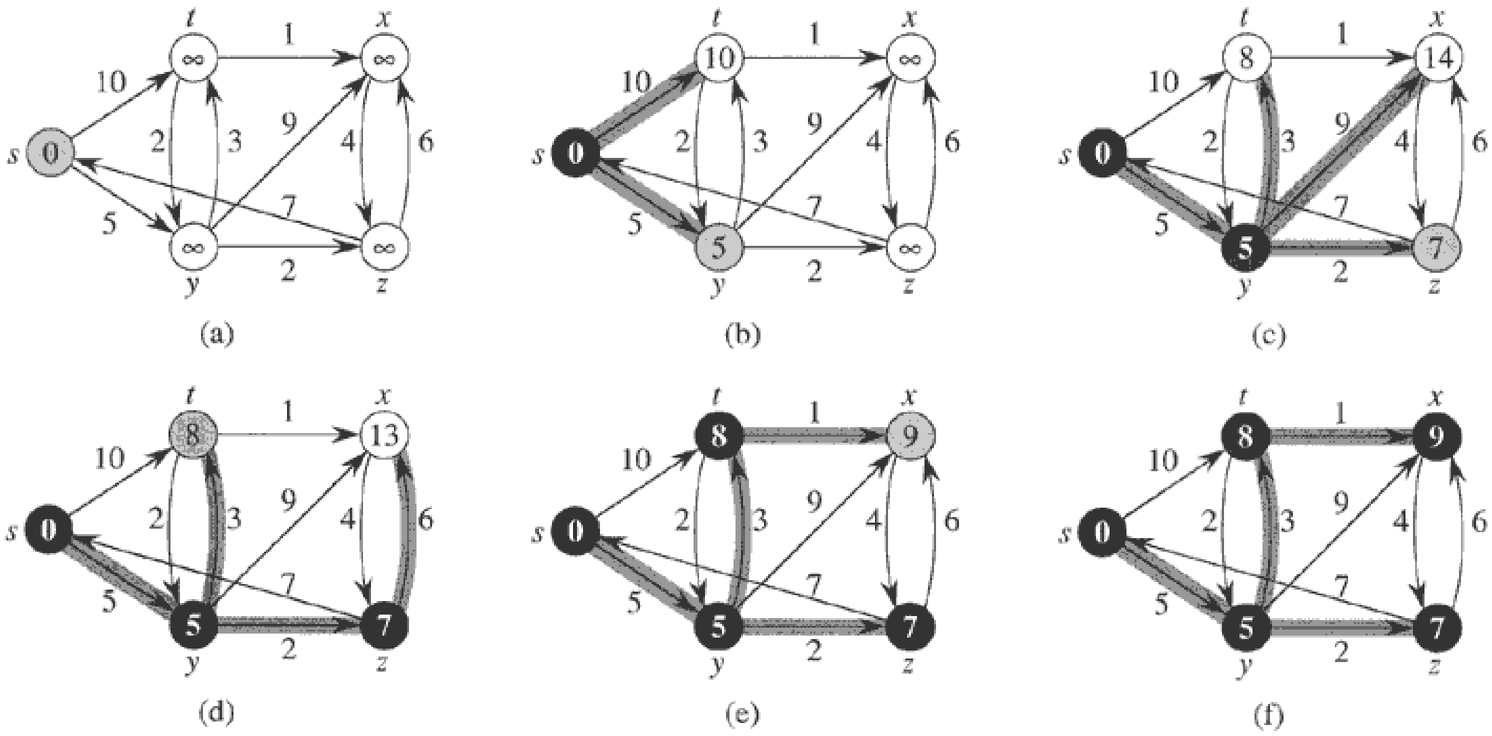
(8.2.2015)

1. האלגוריתם של דייקסטרה (Dijkstra) למציאת מסלול קצר ביותר

האלגוריתם של דייקסטרה פותר את הבעיה של מציאת מסלולים קצרים ביותר ממקור יחיד עבור גרף ממושקל G = (V,E) במקרה שבו כל משקלות הקשתות הם אי-שליליים. כלומר, אנחנו דורשים שיתקיים $w(u,v) \geq 0$ עבור כל קשת $(u,v) \in E$.

האלגוריתם מנהל קבוצה S של קודקודים אשר עבורם נקבעו כבר סופית משקלי המסלולים הקצרים ביותר מן המקור s. האלגוריתם בוחר בכל איטרציה את הקודקוד $u \in V - S$ בעל האומדן המינימלי של מסלול קצר ביותר, מכניס את u ל-S, ומעדכן את כל הקשתות היוצאות מ-u. האלגוריתם ישתמש בתור קדימויות Q.

```
DIJKSTRA(G,s)
    for each vertex v ∈ V
       do v.distance ← ∞
          v.previous ← NULL
    s.distance ← 0
    S ← ∅
    Q ← V
    while Q ≠ ∅
       do u ← EXTRACT-MIN(Q)
          S ← S ∪ {u}
          for each vertex v adjacent to u and not in S
             do if v.distance > u.distance + w(u,v)
                   then v.distance ← u.distance + w(u,v)
                        v.previous ← u
```



ננתח את סיבוכיות האלגוריתם: הזמן לביצוע האתחול (שורות 1-5) הוא $\Theta(V)$. אם תור הקדימויות Q ממומש בתור ערימת מינימום (minimum heap), אז שורה 6 לוקחת גם היא $\Theta(V)$.

כל פעולת EXTRACT-MIN מתבצעת בזמן $\Theta(\log V)$. מכיוון שמתבצעות $|V|$ פעולות כאלה, נקבל $\Theta(V\log V)$.

עדכון השדה distance בשורה 12 דורש לשנות את מיקומו של הקודקוד בתור Q, וזה ייקח $\Theta(\log V)$. מתבצעות לכל היותר $|E|$ פעולות כאלה, בזמן כולל של $\Theta(E\log V)$.

בסך הכל קיבלנו $\Theta((V+E)\log V)$, שהוא $\Theta(E\log V)$.

Study_For_Exam.cpp

```
void Exam_Study() {
    while (knowledge<= max_knowledge) do
    {
        Study(10, minutes, hard);
        while (facebook_feed!=const){
            waste_time();
        }
        if (hungry==true)
            for (i=0;i<10;i++){
                refrigerator.open();
                refrigerator.close();
            }
        else{
            Study(10, minutes, hard);
            hungry=true;
        }
        delay(500);
        Study(10, minutes, NotSoHard);
        delay(1500);
        if (GiveUp==true)
            break;
    } //end while

    Fail_Exam();
    knowledge=0;
}
```

2. פתרון משוואות נסיגה לינאריות הומוגניות

ראינו כיצד פותרים משוואות נסיגה לינאריות הומוגניות. כלומר, משוואות נסיגה מהצורה:

$$f(n) = c_1 \cdot f(n-1) + c_2 \cdot f(n-2) + c_3 \cdot f(n-3) + ... + c_k \cdot f(n-k)$$

בכל הדוגמאות שראינו עד כה, המשוואה האופיינית המתקבלת הייתה משוואה ריבועית. כעת, נראה משוואה לינארית הומוגנית שהמשוואה המאופיינת שלה היא משוואה ממעלה שלישית.

שאלה

נתונה הנוסחה הרקורסיבית עם כלל הנסיגה: $T(n) = 6T(n-1) - 11T(n-2) + 6T(n-3)$. נתונים תנאיי ההתחלה $T(0) = 0$, $T(1) = 1$, $T(2) = 2$. מצאו ביטוי מפורש עבור $T(n)$.

בכל נוסחאות הנסיגה הלינאריות ההומוגניות שפתרנו עד כה, פתרונותיה של המשוואה האופיינית היו שונים זה מזה. מה עושים כאשר פתרון מסוים a של המשוואה האופיינית חוזר על עצמו m פעמים? מתאימים לו את m הביטויים השונים: $a^n, na^n, n^2a^n, ..., n^{m-1}a^n$. נדגים זאת בשאלה הבאה.

שאלה

נתונה סדרת הנסיגה הבאה $a_n = 4a_{n-1} - 4a_{n-2}$, ששני איבריה הראשונים הם $a_0 = 0, a_1 = 1$. מצאו ביטוי מפורש עבור האיבר הכללי a_n בסדרה.

תשובה

נציב α^i במקום a_i , ונקבל: $\alpha^n = 4\alpha^{n-1} - 4\alpha^{n-2}$.

נצמצם ב- α^{n-2} , ונקבל: $\alpha^2 = 4\alpha - 4$.

נעביר אגפים, ונקבל את המשוואה האופיינית: $\alpha^2 - 4\alpha + 4 = 0$.

נפתור את המשוואה הזו לפי נוסחת השורשים, ונקבל: $\alpha_1 = \alpha_2 = 2$.

נשים לב, כי להבדיל מהמצב בשאלות קודמות, כאן למשוואה האופיינית יש שני פתרונות שווים. במקרה כזה, צורת הפתרון היא שונה: $a_n = A_1 \cdot 2^n + A_2 \cdot n \cdot 2^n$

כדי לחשב את ערכם של הקבועים A_1, A_2 נשתמש בתנאי ההתחלה.

נציב $n = 0$ ונקבל: $0 = A_1$

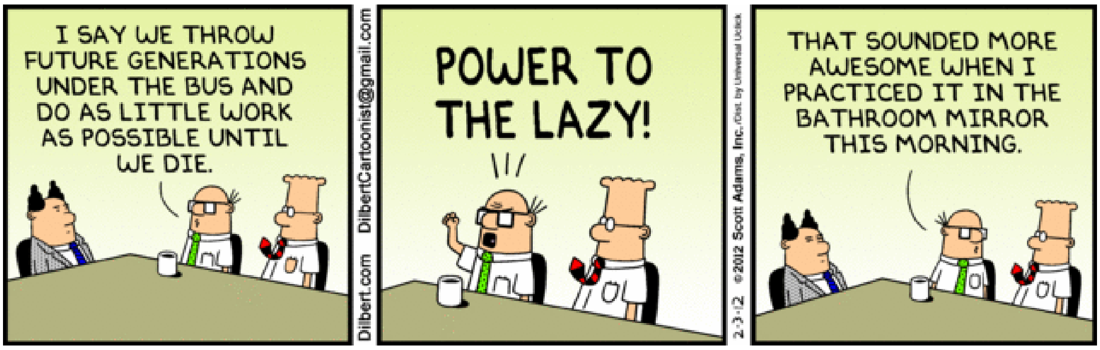
נציב $n = 1$ ונקבל: $1 = 2A_1 + 2A_2$

קיבלנו מערכת של שתי משוואות בשני נעלמים, שפתרונה: $A_1 = 0, A_2 = \frac{1}{2}$.

ואז הביטוי המפורש של נוסחת הנסיגה הוא: $a_n = \frac{1}{2} \cdot n \cdot 2^n$.

לחלופין: $a_n = n \cdot 2^{n-1}$.

מבחינה אסימפטוטית: $a_n = \Theta(n \cdot 2^n)$.



<u>שאלה</u>

נתונה הנוסחה הרקורסיבית עם כלל הנסיגה: $T(n) = 2T(n-1) + 5T(n-2) - 6T(n-3)$. נתונים שלושת תנאיי ההתחלה: $T(i) = i$, לכל $0 \le i \le 2$. מצאו ביטוי מפורש עבור $T(n)$.

<u>שאלה</u>

נתונה הנוסחה הרקורסיבית עם כלל הנסיגה: $T(n) = 5T(n-1) - 3T(n-2) - 9T(n-3)$. נתונים תנאיי ההתחלה $T(0) = 6 , T(1) = -8 , T(2) = -22$. מצאו ביטוי מפורש עבור $T(n)$.

<u>שאלה</u>

נתונה הנוסחה הרקורסיבית עם כלל הנסיגה: $T(n) = 5T(n-1) - 8T(n-2) + 4T(n-3)$. נתונים שלושת תנאיי ההתחלה: $T(i) = i$, לכל $0 \le i \le 2$. מצאו ביטוי מפורש עבור $T(n)$.

<u>שאלה</u>

מסמנים ב-w_n את מספר המילים השונות באורך n, המורכבות רק מהאותיות a, b או c, כך שמתקיימים לגביהן שני התנאים הבאים: (1). המילה נפתחת ומסתיימת באות a ; (2). שתי אותיות סמוכות במילה – שונות זו מזו.

א. חשבו את ערכי w_n עבור $n = 0, 1, 2, 3, 4, 5$.

ב. הסבירו מדוע מתקיימת נוסחת הנסיגה $w_n = w_{n-1} + 2w_{n-2}$.

ג. מצאו פתרון מפורש לנוסחת הנסיגה הלינארית ההומוגנית מסעיף ב', תוך שימוש בתנאי ההתחלה שחושבו בסעיף א'.