

## מבני נתונים ואלגוריתמים – 88-280-02

### תרגיל 2 – מחסניות ותורים

תאריך הגשה: 17/11/2012 (עד 123:50)

#### הוראות הגשה:

יש להגיש את התרגיל דרך האתר – [submit.cs.biu.ac.il](http://submit.cs.biu.ac.il)

יש לציין בתחילת כל קובץ בהערה שם ות.ז.

עבור החלק הראשון יש להגיש קובץ יחיד בשם `targil2_1.c` (למי שמגיש ב-C) או `targil2_2.cpp` (למי שמגיש ב-C++).

עבור החלק השני יש להגיש קובץ יחיד בשם `targil2_2.c` (למי שמגיש ב-C) או `targil2_2.cpp` (למי שמגיש ב-C++).

\* למי שמגיש ב-C++ - מותר להשתמש בספריות STL ☺

#### ניקוד:

מספר הנקודות המקסימלי בבדיקה האוטומטית בכל חלק הוא 50 (סה"כ 100).

#### חלק 1 – פיתרון מבוך באמצעות מחסנית

בתרגיל זה עליכם לממש אלגוריתם שפותר מבוך.

הקלט: זוג מספרים שהם מימדי המטריצה, מטריצה דו-מימדית בינארית (בכל תא מופיע 1 או 0), נקודת התחלה  $(s1, s2)$  ונקודת סיום  $(e1, e2)$  – ראו את הדוגמאות (אפשר להניח שהקלט תקין).

עליכם להגיע מנקודת ההתחלה לנקודת הסיום דרך תאים בהם יש 1 בלבד.

הפלט: שחזור של המסלול – יש לסמן על מטריצת עזר שכולה אפסים את המסלול באמצעות 1-ים. אם לא קיים מסלול כזה הפלט יהיה -1.

- אם קיים מסלול, הוא יחיד (שלא כולל חזרות על איברים).
- כל תו מופרד ברווח יחיד.

הגבלות:

- 1) התוכנית חייבת להשתמש במחסנית שבה יופיע המסלול הרלוונטי.
- 2) במחסנית כל איבר יכול להופיע אך ורק פעם אחת.
- 3) זיכרון: ניתן להשתמש אך ורק במטריצת עזר בגודל המבוך, מטריצת פיתרון בגודל המבוך ובמחסנית (שאינה מוגבלת). ניתן להשתמש במשתני עזר.
- 4) סיבוכיות: זמן הריצה צריך להיות כגודל המטריצה  $O(n*m)$ .

רמז: האלגוריתם צריך לעצור אך ורק כאשר המחסנית ריקה.

## דוגמאות:

```
Input:
4 4
1 0 1 1
1 1 0 0
0 1 1 0
0 0 0 1
0 0
2 2
Output:
1 0 0 0
1 1 0 0
0 1 1 0
0 0 0 0
Input:
5 5
1 1 1 0 0
1 1 1 0 0
1 1 1 0 0
0 0 0 1 1
0 0 0 1 1
0 0
4 4
Output:
-1
```

### **חלק 2: מציאת אורך המסלול הקצר ביותר באמצעות תור**

התרגיל יתבצע על אותו מבוך מהחלק הראשון. עליכם לממש אלגוריתם המשתמש בתור על מנת למצוא את אורך המסלול הקצר ביותר מתחילת המבוך (0,0) עד לציאה כלשהי בקיר הימני.

הקלט: זוג מספרים שהם מימדי המטריצה, מטריצה דו-מימדית בינארית (בכל תא מופיע 1 או 0), נקודת התחלה (s1,s2) ונקודת סיום (e1,e2) – ראו את הדוגמאות (אפשר להניח שהקלט תקין).

הפלט: אורך המסלול הקצר ביותר (סקאלר). אם לא קיים מסלול הפלט יהיה -1.

```
Input:
4 3
1 1 1
1 0 1
1 1 1
0 0
2 0
Output:
3
Input:
3 3
1 1 0
1 1 0
0 0 1
0 0
```

2 2

Output:

-1

\* שימו לב שלא משנה אם יש כמה מסלולים מינימליים, כי הרי הפלט הוא מספר הצעדים המינימלי.

הגבלות:

- 1) זיכרון: יש להשתמש בתור (Queue) ובמטריצת עזר שתשמור עבור כל אחד מהאיברים את מספר הצעדים המינימאלי מנקודת ההתחלה עד אליו.
- 2) סיבוכיות: זמן הריצה צריך להיות כגודל המטריצה  $O(n*m)$ .

על מנת להקל עליכם - אתחלו מטריצת אפסים ובמקום שמייצג את נקודת ההתחלה סמנו 1.  
מבחינתכם בשלב זה, אם נקודת ההתחלה הייתה נקודת הסיום אז מספר הצעדים אליה היה 1.

**בהצלחה!!!**