

**שאלה 3. (30 נקודות)** נתונה מטריצה  $M$  בגודל  $100 \times 100$  שערכי תאיה הינם המספרים 0 או 1. יהי  $k$  מספר שלם שמציין את כמות התאים במטריצה שערכם 1. המטריצה והשלם  $k$  שניהם מצויים ב  $shared$  memory עבור 4 תהליכים. כלומר ישנו struct שממופה ל  $shared$  memory שמוגדר כך

```
struct MAP{
    int k;
    int M[100][100];
};
```

על ארבעת התהליכים לסרוק את המטריצה באופן פסדו-מקבילי על מנת לבקר ביחד בכל התאים שיש בהם את הספרה 1. למען הסר ספק נבהיר שאין חובה שכל תהליך יבקר בכל התאים שערכם 1. הדגש כאן הוא שבמהלך סריקתם המשותפת של המטריצה התהליכים יבקרו באופן קולקטיבי בכל התאים שערכם 1. נפרט עתה כיצד עליכם לממש סריקה זו ואילו אילוצים עליכם לכבד במימוש שאתם מציעים.

הגדרה: זוג תאים שונים  $M[i,j]$  ו  $M[r,k]$  יקראו **סמוכים** אם  $i=r$  ו  $|k-j|=1$  או אם  $k=j$  ו  $|r-i|=1$ . נאמר שתאים סמוכים מגדירים קו מפריד ביניהם ונכנה קו מפריד זה **כצלע**. באופן זה כל תא במטריצה מוקף ב 4 צלעות.

ארבעת התהליכים ממוקמים תחילה בתאים  $(0,0), (0,99), (99,0), (99,99)$ . יהי  $P$  אחד מן התהליכים. על מנת לסרוק את המטריצה  $P$  עובר אך ורק בין תאים סמוכים במטריצה ומחפש אחר תאים שערכם 1. בכל רגע נתון  $t$  אם כך ישנו מסלול של תאים דרכם  $P$  עבר מהתא הראשוני בו זה הושם על מנת להגיע לתא הנוכחי בו זה נמצא בזמן  $t$ . נסמן ב  $Path(P,t)$  את אוסף התאים הללו שמגדירים את המסלול לעיל. היות ו  $P$  עובר רק בין תאים סמוכים נוכל לומר ש  $Path(P,t)$  מגדיר אוסף של צלעות נסמנו ב  $Edge(P,t)$  דרכן עבר  $P$  לאורך המסלול של תאים לעיל על מנת להגיע לתא הנוכחי בו זה נמצא בזמן  $t$ .

על אופן הסריקה לכבד את האילוצים הבאים:

אילוץ 1: אין הגבלה על כמות התהליכים שיכולה לשהות בתא נתון בו זמנית.

אילוץ 2: תהי  $t$  נקודה בזמן. יהי  $P$  תהליך. קבוצת הצלעות  $Edge(P,t)$  הינה אסורה לשימוש לצורכי מעבר לכל שאר התהליכים (שאינם  $P$ ).

אילוץ 3: על התהליכים לתקשר ביניהם על מנת לעדכן אחד את השני בכל פעם שעוד תא עם ערך 1 שלא נמצא קודם לכן נחשף. הינכם רשאים לבחור את מנגנון התקשורת בין התהליכים כאוות נפשכם (דהיינו סיגנלים, Pipes וכיו"ב). בנוסף, הינכם רשאים להגדיר thread אחד (בנוסף ל thread הראשי) לכל תהליך שיעזור לו לעקוב אחרי העדכונים שהתהליך מקבל בזמן שה thread הראשי של התהליך עסוק בסריקה עצמה. מבנה הנתונים שבאמצעותו תעקבו אחרי העדכונים הינו נתון גם לבחירתכם.

אילוץ 4: ברגע שכל התאים בהם יש 1 נמצאו באופן קולקטיבי כל התהליכים מפסיקים את פעולתם והתוכנית מסתיימת.

אילוץ 5: סינכרון רק באמצעות סמפורים.

ענו על הסעיפים הבאים:

**סעיף א (10 נקודות):** תארו את האלגוריתם שברצונכם לממש עבור הסריקה לעיל על כל אילוציה. מספיק לתאר באופן מילולי קצר ביותר את האלגוריתם עבור תהליך בודד. אין צורך בקוד או פסדו-קוד רק לספר במילים שלכם מהו האלגוריתם שאתם מתכוונים לממש בסעיף הבא. הרשו לעצמכם כאן לתת תשובה אבסטרקטית מבלי להיכנס לכל פרט מייגע (אין צורך להסתבך כאן רק להעביר את הרעיונות המרכזיים של האלגוריתם שלכם).

**סעיף ב (20 נקודות):** ממשו את האלגוריתם שהצעתם לעיל ב C (מעל Linux). ספקו רק את הקוד עבור התהליך שמתחיל מתא  $(0,0)$ .

אנא הקפידו על קידוד תקין וקריא. ספקו הערות מלוות לקוד שלכם כמקובל בקהילה על מנת להגביר את קריאות הקוד שלכם. ליד כל משתנה ספקו הסבר מה מטרתו. ראוי לחשוב על פסקה מקדימה לקוד שלכם עם הסברים מדויקים לכל המשתנים שיופיעו בקוד. עליכם לחתור לאלגוריתם בעל פסודו-מקביליות מירבית. כל דבר שאתם רוצים להשתמש בו מחומר הקורס על מנת לפתור את הבעיה הינכם רשאים לעשות בו שימוש כל עוד אף אילוץ לעיל לא מופר.