

קראו את Section 6.3 בסיכומי ההרצאות וענו על השאלה הבאה הלקוחה ממבחן של סמסטר ב 2016. השאלה מובאת עם הניקוד שהתלווה לזו במבחן בו זו הופיעה. יש להגיש העתק אלקטרוני של תשובתכם דרך ה Moodle של המתרגלים.

### שאלה 3: (15 נקודות) Apple and MS cross a river

קבוצה אינסופית של אנשי Apple (המוצלחת והאהובה) וקבוצה אינסופית של אנשי Microsoft עומדים על גדות נהר ומעוניינים לחצות את הנהר. אנו נחשוב על אנשים אלו כעל threads או כעל תהליכים ולכן יש שני סוגים של threads: סוג שמייצג איש Apple וסוג שמייצג איש מחברת Microsoft.

לשם החצייה ישנה רק סירה אחת. סירה זו מבצעת חצייה אחת מגדה ראשונה לשנייה ולא חוזרת שוב. בנוסף ישנם האילוצים הבאים על התנהלות הסירה והחצייה:

- בסירה יש מקום ל 4 אנשים. והסירה לא תעזוב את החוף אם יש עליה פחות או יותר אנשים.
- לאור האיבה בין החברות אסור שהסירה עזוב את החוף אם יושבים בה 3 אנשים מחברה אחת והאדם הרביעי מהחברה השנייה. כל קומבינציה אחרת מתקבלת היא בטוחה.
- כדי לציין thread (כלומר איש) עלה לסירה עליו לקרוא לפרוצדורה board() שמבחינתנו זו פרוצדורה שלא עושה דבר וחוזרת מיד ואנו רק משתמשים בה לסימון העלייה.
- לאחר שארבעה threads (כלומר אנשים) עלו (כלומר קראו ל board()) על הסירה לפי החוקיות לעיל, בדיוק אחד מהם קורא לפרוצדורה rowBoat() כדי לציין שהוא אוחד במשטים של הסירה. thread זה יקרא הקפטן של הסירה לצורך החצייה.
- גם rowBoat() הינה פונקציה לשם סימון שאינכם נדרשים לממש.

ממשו בפסודוקוד את הקוד עבור thread (כלומר איש) מחברת Apple בלבד.

p

❖ הינכם רשאים להתעלם מכיוון החצייה.

❖ הינכם רשאים להשתמש ב barrier synchronisation (ראו חומר מהתרגולים) באופן הבא.

□ השורה barrier = Barrier(n) יוצרת אובייקט Barrier עבור n תהליכים.

□ על מנת להצטרף לאובייקט barrier יש לקרוא למתודה join() שלו כך: barrier.join() אשר

מבטיחה שכל עוד פחות מ n תהליכים קראו ל barrier.join() אזי כל מי שקרא חסום בתוך

ה barrier וכאשר התהליכון האחרון מבצע barrier.join() אזי זה משחרר את כולם.

□ אין צורך לממש Barrier synchronisation זה ניתן לכם בחינם.

◆ אין צורך לממש את board() או את rowBoat()

◆ פרט ל Barrier synchronisation שנתון לכם בחינם כל שאר הסינכרון חייב שיבוצע על ידי סמפורים בלבד.