



Operating Systems

Assignment Replacing Course Exam

Dr. Moshe Deutsch

mdeuts@ruppin.ac.il

Submission Instructions:

1. במשימה שאלה אחת. על מימוש השאלה להיות ב C בסביבת Win32 API. יש להגיש (בהתאם להוראות ההגשה להלן) רק קבצי מקור (קבצי c. וקבצי h. אם קיימים) ולא קבצי פרוייקט ב VS.

2. חובה לכתוב את המטלה בסביבת VS2019 בלבד!

3. הקוד חייב להיות Well-Commented!!!.

4. כפי שתראו בהמשך תיאור המטלה, תצטרכו ליצור 2 פרוייקטים נפרדים ב VS עבור כל אחד מ 2 ה Processes הנדרשים: פרוייקט אחד עבור "נמל חיפה" ופרוייקט שני עבור "נמל אילת". לפיכך:

a. יש לתת שמות משמעותיים לקבצי המקור (בהתאם לתפקידם!) – לדוגמא: EilatPort.c, HaifaPort.c.

b. יש לפתוח שתי תתי ספריות HaifaSrc ו EilatSrc, לקבצי המקור של כל פרוייקט עבור "נמל חיפה" ועבור "נמל אילת" (בהתאמה!). בכל אחת מן הספריות הללו, יש לשים את כל קבצי המקור של הפרוייקט המתאים (קבצי c. וקבצי h. אם קיימים).

c. בנוסף, יש להעתיק את כל התוכניות במלואן לקובץ WORD אחד. בקובץ זה, עבור כל קובץ מקור שמועתק לשם, יש לציין את שם קובץ המקור הממומש ותפקידו (אם השתמשתם בקבצי h. יש להעתיקם גם כן לקובץ ה WORD). סדר הדברים:

i. בתחילת הקובץ הנ"ל, יש לציין את שמות המגשים (שם + משפחה לכל אחד) ות.ז. שלהם (במודגש!).

ii. לאחר מכן, יש להעתיק לקובץ הנ"ל את כל קבצי המקור של "נמל חיפה".

iii. אח"כ, יש להעתיק לקובץ הנ"ל את כל קבצי המקור של "נמל אילת".

iv. בנוסף, בסוף הקובץ הנ"ל, יש לצרף כמה דוגמאות מייצגות לפלט ממסך ההרצה.

שם קובץ ה-WORD יהיה כך: OS_FinalProj_GroupX. כאשר: X מציין את המספר הסידורי של צוות ההגשה, כפי שמופיע בטבלה הסופית של "חלוקה לצוותים" באתר הקורס (המספר הסידורי של כל צוות מופיע בעמודה A בקובץ: "חלוקה לצוותים" – לינק לקובץ זה נמצא בחלק הימני העליון של אתר הקורס). שם קובץ ההגשה יהיה באנגלית בלבד!

5. יש לשים את כל ה sources בתתי הספריות HaifaSrc ו EilatSrc (עפ"י המוסבר בסעיף 4b למעלה ואת קובץ ה WORD מעל שתי הספריות הללו, במבנה הספריות, ולכוון לתוך

קובץ ZIP או RAR **אחד** את כל אלה ולהעלותו לאיזור המטלה באתר הקורס. שם קובץ ה-RAR/ZIP יהיה כך: OS_FinalProj_Group**X**. כאשר: **X** מציין את המספר הסידורי של צוות ההגשה, **כפי שמופיע בטבלה הסופית של "חלוקה לצוותים" באתר הקורס** (המספר הסידורי של כל צוות מופיע בעמודה A בקובץ: "חלוקה לצוותים" – לינק לקובץ זה נמצא בחלק הימני העליון של אתר הקורס). **שם קובץ ההגשה יהיה באנגלית בלבד!**

6. כאשר פותחים את הקובץ המכוון, יש לקבל 3 דברים: קובץ ה WORD (סעיף 4c), ספריה HaifaSrc (המכילה את כל ה source files של "נמל חיפה" – סעיף 4b) וספריה EilatSrc (המכילה את כל ה source files של "נמל אילת" – סעיף 4b).

7. תאריך אחרון להגשת המטלה: מוצ"ש, תאריך 14/08/2021 – עד השעה 23:59 בלילה. לא ניתן לאחר בהגשה! מעבר לתאריך זה, המערכת תיסגר באופן אוטומטי להגשות ומי שלא יגיש עד לתאריך זה (בשעה הנקובה) ייקבל באופן אוטומטי ציון סופי 0 במטלה.

8. יש להגיש מטלה זו בקבוצות אליהן שובצתם, עפ"י טבלת השיבוצים הסופית, כפי שמופיעה באתר הקורס - **רק סטודנט אחד מהקבוצה המגישה יעלה את המשימה לאתר ה-Moodle, כנציג הקבוצה.**

9. **אזהרה חמורה:** מטלות שתוגשנה לא בהתאם להוראות ההגשה לעיל, תספוגנה הורדה **ניכרת בציון הסופי!**

* אזהרה:

המרכז אקדמי רופין רכש לאחרונה מערכת ממוחשבת לזיהוי העתקות. כל הגשותיכם ייבדקו ע"י המערכת הזו. כל הגשה, לגביה תהיה התראת העתקה – ולו הקטנה ביותר – תיפסל באופן אוטומטי וכל הסטודנטים הנוגעים בדבר יועלו באופן מיידי לוועדת משמעת!

Exercise – 100%

מימוש סימולציה למעבר דבוקת כליי שייט (Cluster of Vessels) בתעלת סואץ – בשפת C, בסביבת Win32 API.

1. סיפור המעשה:

1.1. תעלת סואץ היא תעלת מים מלאכותית, העוברת במצרים, ממערב למדבר סיני, ומשמשת למעבר אוניות: ממזרח אסיה, האוקיינוס ההודי והים האדום – לים התיכון ולאירופה. אורכה 162.5 ק"מ והיא נמתחת בין הערים פורט סעיד ופורט פואד בצפון, לבין העיר סואץ בדרום, ולאורכם של שלושה ממחוזות מצרים: פורט סעיד, אסמאעיליה וסואץ. רוחבה של התעלה נע בין 250 ל-500 מטרים, ועומקה בין 10 ל-20 מטרים.

1.2. חשיבותה הגדולה של התעלה היא מתן האפשרות לספינות הנעות בין אסיה לאירופה להימנע מהצורך להקיף את אפריקה דרך כף התקווה הטובה והאוקיינוס האטלנטי הסוער, ובכך לקצר ולהוזיל משמעותית את הדרך.

1.3. התעלה מהווה כיום את קו הגבול בין יבשת אסיה ליבשת אפריקה. בין מלחמת ששת הימים (1967) ועד מלחמת יום הכיפורים (1973) הייתה התעלה גם קו הגבול בין ישראל למצרים. ישראל עזבה את האזור בהסכם הפרדת הכוחות בין ישראל

למצרים שנחתם ב 1974. ב 6-באוגוסט 2015 נחנכה תעלת סואץ החדשה המקבילה לתעלה הקיימת ואורכה 72 ק"מ – בכך מתאפשרת תנועה דו-סטריית של אוניות, כאשר כל אחת מן התעלות מהווה נתיב בכוון מסויים.

1.4. להלן תמונה, הממחישה את החיבור של תעלץ סואץ לים התיכון – והמשכה לכוון מפרץ סואץ:



1.5. להלן תמונה, הממחישה את הקונספט של שתי תעלות, המאפשרות, יחדיו, תנועה דו סטריית של כליי שיט (Vessels):



1.6. הסימולציה עוסקת בדבוקה (Cluster) של N כללי שייט (Vessels) שיוצאים מנמל חיפה (שנמצא בים התיכון), ע"מ לפרוק סחורה בנמל אילת (שנמצא בים האדום), ומייד לאחר מכן לחזור בחזרה לנמל חיפה. המעבר בין הים התיכון לים האדום (ובחזרה!) מתבצע דרך תעלת סואץ. * מספר כלי השייט (N) הינו בר שינוי, לכן לא ננקב מספר עקרוני של כללי שייט – פרטים נוספים בהמשך.

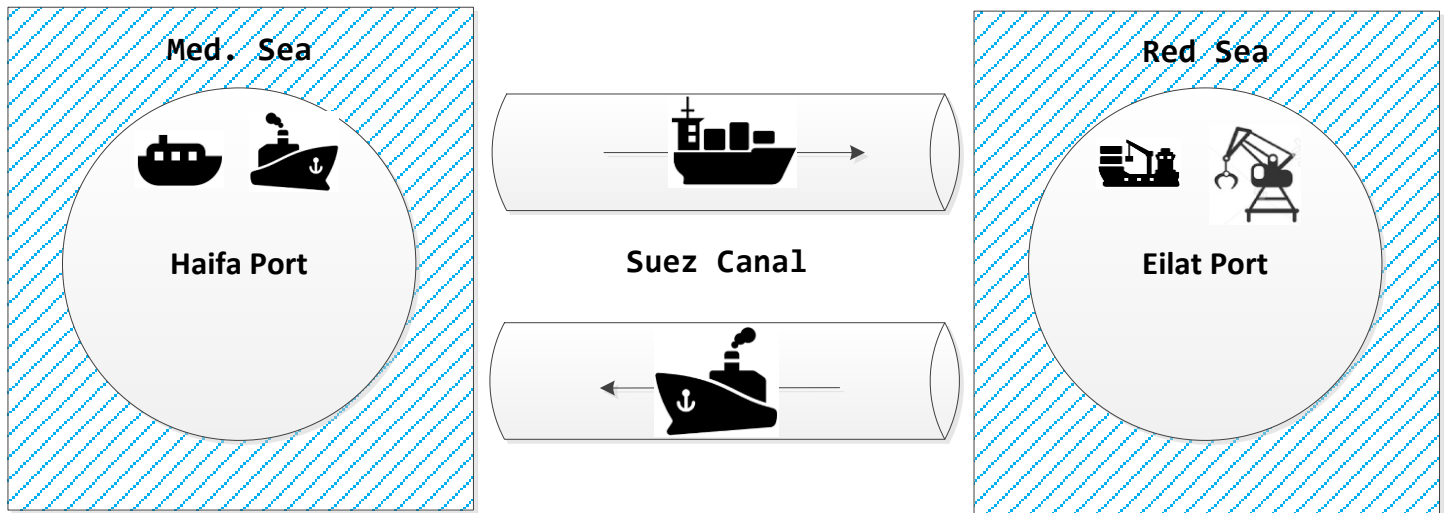
1.7. למעשה, כל כלי שייט מפליג מנמל חיפה לנתיב תעלת סואץ שמוביל מהים התיכון לים האדום; משם מגיע (דרך מיצרי טיראן) לנמל אילת; נכנס לרציף הפריקה (Unloading Quay) בנמל אילת; ברציף הפריקה, מוצמד לכל כלי שייט מנוף (שמוקדש לו!) שפורק ממנו את הסחורה; מייד לאחר גמר הפריקה, כלי השייט שנפרק עושה את דרכו בהפלגה מנמל אילת לתוך נתיב תעלת סואץ בכוון הנגדי – שמוביל מהים האדום לים התיכון ומשם בחזרה לנמל חיפה.

1.8. להלן תמונה הממחישה את המסלול:



2. סמנטיקה: מיפוי הקונספטים בסימולציה לעולם המקביליות:

2.1. נמל חיפה ונמל אילת הם Processes; שני המעברים בתעלת סואץ (כאשר, כל מעבר הוא חד כווני בכוון מסויים) הם Anonymous Pipes; כללי השייט השונים הם, כמובן, Threads; רציף הפריקה בנמל אילת (Unloading Quay) הוא, למעשה, ADT שהגישה אליו מווסתת ע"י Barrier (הקונספט של Barrier ופרטי תהליך הפריקה ייפורטו בפרק 5 בהמשך); המנופים של רציף הפריקה הם Threads. הרעיון המרכזי (בהקשר מיפוי הקונספטים לעולם המקביליות) מתואר בתרשים הבא:



2.2. בסעיפים הבאים של פרק זה, נתאר את הקונספטים המרכזיים בסימולציה. פיירוט נוסף יינתן בפרקים שבהמשך.

2.3. כללי השייט השונים, נוצרים בנמל חיפה – מספר כללי השייט מתקבל בשורת הפקודה (ייפורט בפרק 3 בהמשך). לכל כלי שייט יש מזהה ייחודי (Unique ID) מספרי. ה ID's הללו ניתנים באופן סדרתי וחייבים להתחיל מ 1.

2.4. כזכור, ב Anonymous Pipes ניתן להעביר רק מידע, כך שלא ניתן לבצע מעבר פיזי של כללי שייט מנמל הבסיס לנמל היעד (ובחזרה!). לפיכך, הרעיון המרכזי הוא להגדיר פרוטוקול עבודה (עפ"י הפרטים הנתונים במסמך זה) שייאפשר לסמלץ את הפונקציונאליות הנדרשת בשני הנמלים, כאשר מה שנשלח באמצעות כל Pipe הוא מידע לגבי כלי השייט שעובר (בכל פעם). כאשר בקצה של Pipe מסויים, מתקבל מידע על מעבר כלי שייט כלשהו לתוך הנמל – על המידע הזה לשמש, באופן הכי יעיל, את הנמל המקבל, בכדי לסמלץ את כלי השייט הזה, בהתאם לפונקציונאליות הנדרשת/מוגדרת במסמך זה.

2.5. כל נתיב בתעלת סואץ יכול להכיל בזמן נתון רק כלי שייט אחד. מאחר ושני נתיבי התעלה הם בכוונים שונים ניתן (וחובה לאפשר זאת, מבחינת נצילות המשאבים!) שיהיו בזמן נתון שני כללי שייט, הנעים בכוונים שונים בשני נתיבי התעלה. המשמעות היא ששני נתיבי התעלה אינם תלויים זה בזה.

2.6. מי שאחראים "להאזין" למידעים המתקבלים לגבי הגעה של כליי שייט לכל נמל הם ה main Threads של שני הנמלים (כל אחד בצידו הוא, כמובן).

2.7. הפונקציה ReadFile לקריאה של מידע בינארי מקצה של Anonymous Pipe כלשהוא – הינה Blocking Function. כלומר, Thread שמנסה לבצע קריאה מה Pipe ואין ב Pipe מידע, נכנס לתור ההמתנה על משאב זה, לפניי סיום הפו'. כאשר מתקבל מידע בקצה השני של ה Pipe, ה Thread הממתין משתחרר ל Ready Queue וכשיגיע תורו לקבל CPU, ייקרא את המידע מה Pipe ורק אז הפו' ReadFile מסתיימת. נקודה זו חשובה ביותר, לקיום חוסר התלות בין שני נתיבי התעלה, המתואר בסעיף 2.5.

2.8. כל כלי שייט (בכל נמל) אחראי באופן עצמאי לעבור בנתיב המתאים בתעלה (קרי, לתקשר עם ה Anonymous Pipe המתאים), כאשר הוא נדרש לכך בהתאם לפונקציונאליות הנדרשת/מוגדרת.

2.9. בנמל חיפה, לאחר שכלי שייט כלשהו עבר בתעלה – בנתיב המוביל לים האדום (קרי, שלח ל Anonymous Pipe המתאים את המידע, המציין את העובדה שהוא עובר בתעלה), כלי השייט הנ"ל חייב להכנס למצב המתנה (WAIT).

2.10. כאשר ה main Thread של נמל אילת מקבל אינדיקציה על הגעה של כלי שייט כלשהו בתעלת סואץ – בנתיב המוביל לים האדום, על ה main Thread הנ"ל ליצור Thread חדש שייצג, בצידו של נמל אילת, את כלי השייט המגיע – וזאת עפ"י המידע המדוייק שהועבר בנתיב התעלה (המוביל לים האדום) בנוגע לכלי שייט זה. מרגע יצירתו בנמל אילת, כל כלי שייט מבצע (בצד של נמל אילת) את תפקידו בצורה אוטונומית, בהתאם לפונקציונאליות הנדרשת/מוגדרת, עד שמסיים את תפקידו – לפני סיום תפקידו מעביר כלי השייט אינדיקציה על מעבר שלו בתעלה (בנתיב המוביל לים התיכון) בחזרה לנמל חיפה.

2.11. בהמשך לסעיף 2.9, כאשר ה main Thread של נמל חיפה מקבל אינדיקציה על חזרה של כלי שייט כלשהו בתעלת סואץ – בנתיב המוביל לים התיכון, על ה main Thread הנ"ל לבצע Signal ולשחרר באופן ישיר (ובלעדי!) את כלי השייט המתאים (שנמצא במצב

WAIT – ראו סעיף 2.9), ע"מ שכלי שייט זה יוכל להמשיך באופן עצמאי את פעילותו בנמל חיפה, בהתאם לפונקציונאליות הנדרשת/מוגדרת.

3. נמל חיפה:

3.1. על ה Process של נמל חיפה לקבל את מספר כליי השייט ליצירה, באמצעות שורת הפקודה. יש לוודא תקינות מספר זה: חייב להיות ערך מספרי בין 2 ל 50. במידה והערך לא תקין, יש להוציא הודעת שגיאה משמעותית למשתמש ולסיים את הסימולציה.

3.2. ה main Thread של נמל חיפה אחראי לכל האתחולים הנדרשים, כמו גם ליצירת כל כליי השייט (עפ"י המספר שהתקבל משורת הפקודה). כמו כן, הוא, כמובן, אחראי על כל הסגירות ושחרורי הזיכרון, כאשר הוא בטוח בוודאות שכל כליי השייט סיימו את תפקידם גם בנמל אילת וגם בנמל חיפה. עם זאת, כפי שצויין גם בפרק 2, יש ל main Thread של נמל חיפה עוד תפקידים, כדלהלן:

3.2.1. לפני תחילת המעברים של כליי השייט בתעלה ה main Thread של נמל חיפה חייב לקבל אישור מנמל אילת לביצוע המעברים. לשם כך, הוא חייב להעביר, מראש, לנמל אילת (באמצעות התעלה בכוון הים האדום) את המספר של כליי השייט שעתידיים לעבור בתעלה לנמל אילת. לאחר מכן, ה main Thread צריך להמתין לתשובתו של נמל אילת. אם התשובה חיובית, הסימולציה ממשיכה; אם התשובה שלילית, יש להוציא הודעת שגיאה משמעותית למשתמש ולסיים את הסימולציה (הסיבות האפשריות לתשובה שלילית/חיובית מנמל אילת ייפורטו בפרק 4).

3.2.2. ה main Thread של נמל חיפה אחראי "להאזין" למידעים המתקבלים לגבי כליי שייט (שיצאו קודם לכן בתעלה בכוון שמוביל מהים התיכון לים האדום – לנמל אילת וסיימו שם את כל התהליך), שחזרו לנמל חיפה (דרך נתיב תעלת סואץ בכוון שמוביל מהים האדום לים התיכון): כאשר ה main Thread של נמל חיפה מקבל אינדיקציה על חזרה של כלי שייט כלשהו בתעלת סואץ – בנתיב המוביל לים התיכון, על ה main Thread הנ"ל לבצע Signal ולשחרר באופן ישיר (ובלעדי!) את כלי השייט המתאים (שנמצא במצב WAIT – ראו סעיפים: 2.9, 3.3.4), ע"מ שכלי שייט זה יוכל להמשיך באופן עצמאי את פעילותו בנמל חיפה, בהתאם לפונקציונאליות הנדרשת/מוגדרת.

3.3. כללי השייט בצד של נמל חיפה:

3.3.1. כל כלי השייט בנמל חיפה הם מאותה משפחה.

3.3.2. לכל כלי שייט יש מזהה ייחודי (Unique ID) מספרי. ה ID's הללו ניתנים באופן סדרתי וחייבים להתחיל מ 1. ה ID של כל כלי שייט מועבר לו כפרמטר בעת יצירתו.

3.3.3. מרגע יצירתו, כל כלי שייט אחראי באופן עצמאי לעבור בנתיב תעלת סואץ, שמוביל מהים התיכון לים האדום.

3.3.4. לאחר שכלי שייט כלשהו עבר בתעלה – בנתיב המוביל לים האדום (קרי, שלח ל Anonymous Pipe המתאים את המידע, המציין את העובדה שהוא עובר בתעלה), כלי השייט הנ"ל חייב להכנס למצב המתנה (WAIT).

3.3.5. לאחר שכלי השייט קיבל Signal (כתוצאה מהתהליך המתואר בסעיפים: 2.11, 3.2.2) וממשיך לרוץ – הוא מסיים את פעילותו, מייד לאחר מכן.

3.3.6. * חשוב לציין שיש מקומות שונים בהם כל כלי שייט נדרש להוציא הדפסות ל console, כאשר בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי השייט והשלב בו הוא נמצא. פירוט לגבי ההדפסות (הפורמט, השלבים הנדרשים ועוד) יינתן בפרק 6.

3.3.7. * בהמשך לסעיף 3.3.6, חשוב לציין שישנם שלבים מסויימים בהם כל כלי שייט נדרש להגריל זמן שינה (בטווח מסויים) ולבצע Sleep לפרק זמן זה, גם ע"מ להאט את קצב הסימולציה וגם ע"מ לסמלץ סיטואציות שלוקחות זמן מסויים. לדוגמא, לסמלץ שחציית התעלה ע"י כלי שייט מסויים לוקחת זמן מסויים. פירוט לגבי ה Sleep (טווח הזמן, ההגרלה, השלבים הנדרשים ועוד) יינתן בפרק 6.

4. נמל אילת:

4.1. מייד לאחר שה Process של נמל אילת נוצר, ה main Thread שלו צריך לקבל מנמל חיפה (באמצעות נתיב התעלה המוביל בכוון מהים התיכון לים האדום) את המספר של כללי השייט שעתידים לעבור בתעלה לנמל אילת (ראו סעיף 3.2.1). המספר הזה חייב להיות מספר שאינו ראשוני (ע"מ לבנות כשורה את רציף הפריקה – Unloading Quay – פרטים נוספים בהמשך). אם המספר אינו ראשוני, על ה main Thread לשלוח בחזרה לנמל חיפה תשובה חיובית (קרי, אישור מעבר) ולהמשיך בהתאם לפונקציונאליות הנדרשת/מוגדרת; אם המספר ראשוני, על ה main Thread לשלוח בחזרה לנמל חיפה תשובה שלילית ולסיים את ה Process של נמל אילת – בצורה מסודרת!

4.2. אחריותו ותפקידיו הנוספים של ה main Thread של נמל אילת:

4.2.1. הוא, כמובן, אחראי לכל האתחולים הנדרשים.

4.2.2. הוא אחראי לבניית כל המרכיבים הנדרשים למימוש ה Barrier וה ADT של רציף הפריקה (Unloading Quay) – הקונספט של רציף הפריקה יוסבר בפרוטרוט בפרק 5 – מה שייאפשר לכם לתכנן במפורט את המרכיבים הנדרשים למימוש קונספט זה. בכל אופן, המרכיב הבסיסי בקונספט הזה הוא מספר מנופי הפריקה, המשמשים לפריקת המטענים מכלי השייט שנמצאים ברציף הפריקה. זהו גם מספר כללי השייט שאמורים להשתחרר בו-זמנית מראש תור ההמתנה של ה Barrier (שמוסת את כניסת כללי השייט לרציף הפריקה), ברגע שרציף הפריקה מתפנה (* וזאת בתנאי, כמובן, שמספר כללי השייט שממתינים ב Barrier, בנק' הזמן הזה, הוא לכל הפחות כמספר המנופים ברציף הפריקה – פרטים נוספים בפרק 5). על מנת להתחיל לבנות את רציף הפריקה, על ה main Thread להגריל את מספר המנופים ברציף: על מספר זה להיות גדול מ 1 וקטן ממספר כללי השייט, שצפויים להגיע (דרך נתיב התעלה בכוון הים האדום) מנמל חיפה; כמו כן, מספר זה חייב להיות מחלק (Divisor) של מספר כללי השייט הצפויים.

4.2.3. בהמשך לתיאור בסעיפים: 2.4, 2.6 – 2.11 ו 3.3.4, ה main Thread של נמל אילת אחראי "להאזין" למידעים המתקבלים לגבי כללי שייט (שיצאו קודם לכן מנמל חיפה, בתעלה בכוון שמוביל מהים התיכון לים האדום), ש "מגיעים" לנמל אילת (ראו

סעיף 2.10). ברגע שמתקבל, בנתיב התעלה בכוון הים האדום, מידע לגבי הגעת כלי השיט מסויים, על ה main Thread ליצור Thread מתאים, ע"מ לייצג את כלי השיט הזה בנמל אילת – ה ID של ה Thread הנוצר חייב להיות אותו ID של כלי השיט המקורי (בנמל חיפה), שדיווח מעבר בתעלה לכוון נמל אילת (סעיף 3.3.4). * זיכרו: כלי השיט ש "עבר" בנתיב התעלה מנמל חיפה לנמל אילת נכנס למצב WAIT בנמל חיפה (סעיף 3.3.4) וכעת יש לו Thread מייצג בנמל אילת, וכאשר כלי השיט הנ"ל יסיים את כל מה שנדרש ממנו בנמל אילת ידווח מעבר בתעלה "בחזרה" (בנתיב המוביל לים התיכון – קרי, ישלח ל Anonymous Pipe המתאים את המידע, המציין את העובדה שהוא עובר בתעלה), יחל, בנמל חיפה, התהליך המתואר בסעיפים 2.11 ו 3.2.2, בעוד הוא מסיים את חייו בנמל אילת. זוהיא סימולציית המעבר במלוא הדרה!

4.2.4. ה main Thread של נמל אילת אחראי, כמובן, על כל הסגירות ושחרורי הזיכרון של כל האלמנטים בנמל אילת, וזאת רק כאשר הוא בטוח בוודאות בשני דברים שהתבצעו:

- i. שכל כליי השיט, הנמצאים בנמל אילת, סיימו את תפקידם בנמל זה;
- ii. לאחר שהוא בטוח בוודאות ב (i), הוא חייב לדאוג לכך שמנופי הפריקה, ברציף הפריקה (Unloading Quay), יסיימו את חייהם; ולוודא שהם אכן סיימו.

4.3. כללי השיט בצד של נמל אילת:

4.3.1. כל כליי השיט בנמל אילת הם מאותה משפחה.

4.3.2. לכל כלי שיט יש מזהה ייחודי (Unique ID) מספרי, שניתן לו, בעת יצירתו, ע"י ה main Thread, עפ"י הנתונים שהתקבלו באינדיקציה של המעבר בנתיב התעלה – בכוון המוביל לים האדום (ראו סעיפים 2.10, 4.2.3).

4.3.3. מרגע יצירתו, כל כלי שיט, בנמל אילת, אחראי לבצע את תפקידו באופן עצמאי, בהתאם לדרישות כדלהלן:

- i. להכנס לרציף הפריקה (Unloading Quay), עפ"י חוקיות ההגדרות/דרישות, המפורטות בפרק 5;

ii. מייד לאחר סיום פריקת המטען שלו ברציף הפריקה ויציאתו מרציף הפריקה

(פירוט יינתן בפרק 5), על כלי השייט לדווח מעבר בתעלה "בחזרה" (בנתיב

המוביל מהים האדום לים התיכון) לנמל חיפה;

iii. מייד לאחר (ii), על כלי השייט לסיים את תפקידו בנמל אילת.

4.3.4. * חשוב לציין שיש מקומות שונים בהם כל כלי שייט נדרש להוציא הדפסות ל

console, כאשר בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי

השייט והשלב בו הוא נמצא. פירוט לגבי ההדפסות (הפורמט, השלבים הנדרשים ועוד)

יינתן בפרק 6.

4.3.5. * בהמשך לסעיף 4.3.4, חשוב לציין שישנם שלבים מסויימים בהם כל כלי שייט

נדרש להגריל זמן שינה (בטווח מסויים) ולבצע Sleep לפרק זמן זה, גם ע"מ להאט את

קצב הסימולציה וגם ע"מ לסמלץ סיטואציות שלוקחות זמן מסויים. לדוגמא, לסמלץ

שחציית התעלה ע"י כלי שייט מסויים לוקחת זמן מסויים. פירוט לגבי ה Sleep (טווח

הזמן, ההגרלה, השלבים הנדרשים ועוד) יינתן בפרק 6.

5. רציף הפריקה (Unloading Quay) של נמל אילת:

5.1. ראשית, להלן הגדרת הקונספט הבסיסי של Barrier והסברו (לאחר מכן, נציג מודל סכמטי

של רציף הפריקה ונסביר אילו אדפטציות (התאמות) של הקונספט נעשו ב Barrier,

שמווסת את את כניסת כליי השייט לרציף הפריקה):

תבנית נפוצה של תכנות מקבילי (A common Concurrent programming pattern),

הנמצאת בשימוש בעיקר במערכות שליטה ובקרה (Control Systems) ומערכות Reactive

Systems, היא לבצע processing של שלבים עצמאיים ובלתי תלויים: בכל שלב, כל ה

Threads עובדים באופן עצמאי ובלתי תלוי; כלומר, אין כל שיתוף פעולה בין ה Threads ולכן אין

כל סינכרון (synchronization) בין ה Threads במשך כל שלב כזה. אבל, בסוף כל שלב, ה

Threads חייבים לבצע סינכרון (synchronization) **בנקודת סינכרון (synchronization)**

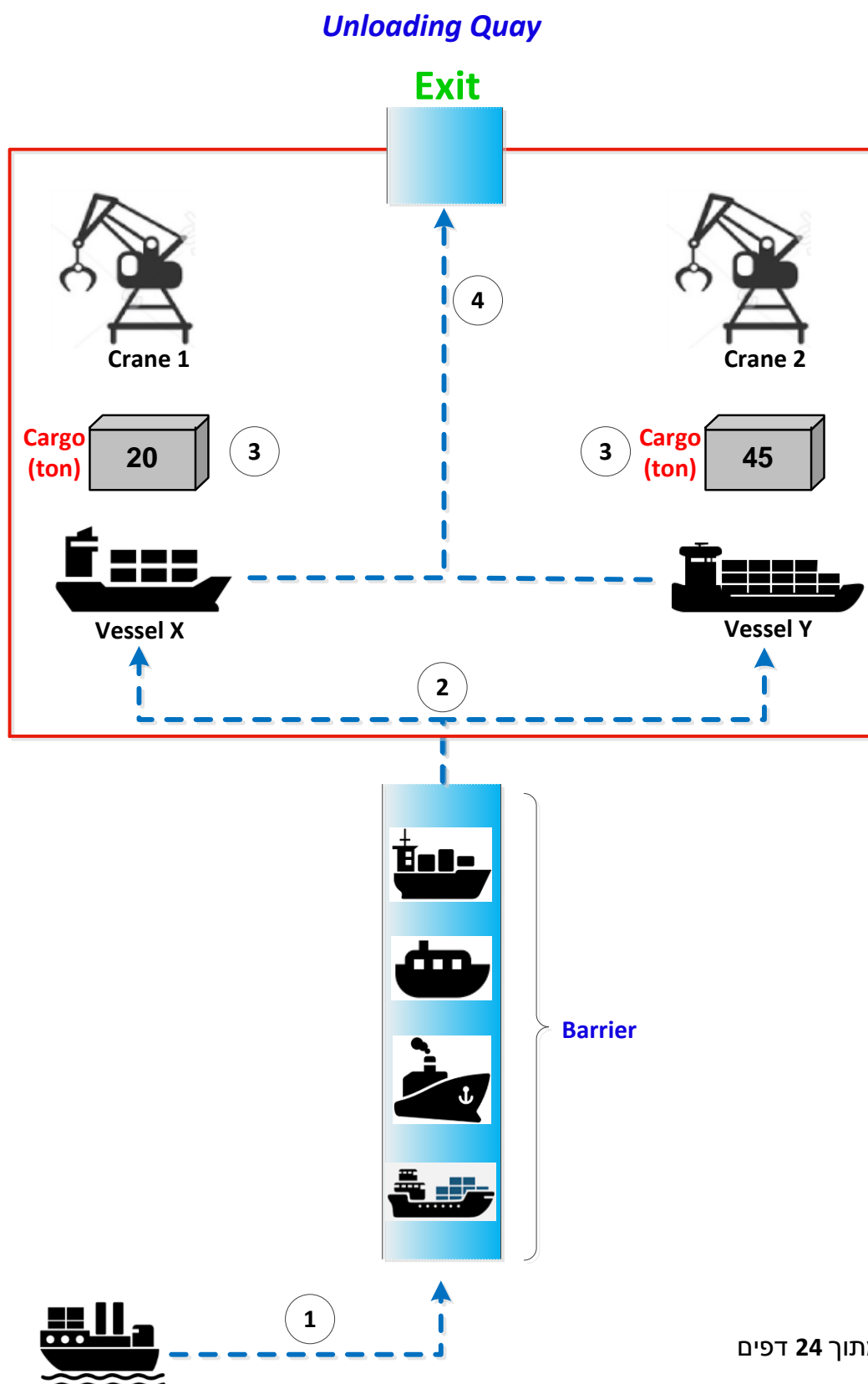
(point), הנקראת **Barrier** (מחסום). כך ש, אם Thread כלשהו מגיע ל **Barrier** לפני ש **כל** ה

Threads האחרים הגיעו אליו, הוא **מחכה (WAIT)**. רק כאשר **כל** ה Threads הגיעו ל

Barrier, הם מקבלים "signal" (כלומר, they are notified) ורק אז כל אחד מהם יכול להתחיל

את ה execution של השלב הבא (שוב, השלב הבא עבור כל אחד מה Threads מתבצע גם כן באופן עצמאי ובלתי תלוי, עד שכל אחד מגיע ל **Barrier** הבא, מחכה לכל האחרים ורק כשכולם הגיעו ל **Barrier** הבא, הם משוחררים... וכן הלאה...).

5.2. להלן מודל סכמטי של רציף הפריקה:



5.3. כאמור בסעיף 4.2.2, על מנת להתחיל לבנות את רציף הפריקה, על ה main Thread של

נמל אילת להגריל את מספר המנופים ברציף: על מספר זה להיות גדול מ 1 וקטן ממספר כליי

השייט, שצפויים להגיע (דרך נתיב התעלה בכוון הים האדום) מנמל חיפה; כמו כן, מספר זה

חייב להיות מחלק (Divisor) של מספר כלי השייט הצפויים. עפ"י מספר זה נבנים:

i. גם ה Threads המייצגים את המנופים/עגורנים (Cranes);

ii. גם הקונספט של ה Barrier;

iii. וגם ה ADT של רציף הפריקה (מוקף במרובע אדום בסכמה לעיל).

התרשים לעיל, מתאר מצב בו מספר כלי השייט (Vessels), המגיעים מנמל חיפה, הוא 4

ומספר המנופים/עגורנים (Cranes), המוגרל, הוא 2.

5.4. המסלול הקונספטואלי שכל כלי שייט עושה ברציף הפריקה מסומן בסכמה לעיל בקו

מקווקו בצבע תכלת, כאשר השלבים השונים ממוספרים באופן ברור (מוקפים בעיגולים

שחורים).

5.5. בסעיפים הבאים ייפורטו הנדבכים השונים, המרכיבים את הקונספט של רציף הפריקה.

הפירוט יתחיל במנופים/עגורנים (Cranes); לאחר מכן ייפורט ה ADT של רציף הפריקה

ולבסוף ייפורט ה Barrier.

5.6. המנופים/עגורנים (Cranes):

5.6.1. כל המנופים/עגורנים הם Threads מאותה משפחה.

5.6.2. לכל מנוף/עגורן (Crane) מזהה ייחודי (Unique ID) מספרי. ה ID's הללו ניתנים

באופן סדרתי וחייבים להתחיל מ 1. ה ID של כל Crane מועבר לו כפרמטר בעת

יצירתו (ע"י ה main Thread).

5.6.3. המנופים חיים למשך כל חיי ה Process של נמל אילת: נוצרים ע"י ה main

Thread באתחול נמל אילת (ראו סעיף 4.2.2) ומסיימים את חייהם כשמקבלים

אינדיקציה מה main Thread על סגירה (ראו סעיף 4.2.4).

5.6.4. כאשר Crane לא מבצע עבודת פריקה, הוא חייב להמצא במצב המתנה (WAIT).

5.7. ה ADT (מוקף במרובע אדום בסכמה לעיל):

5.7.1. מי שמזין את ה ADT הוא ה Barrier (ייפורט בסעיף 5.8). העיקרון הוא, למעשה, וויסות מדורג של הכנסת כללי שייט, ל ADT של רציף הפריקה, בקבוצות של M כללי שייט בכל פעם; כאשר M הוא מספר המנופים ברציף. כלומר, הכנסה בו-זמנית של M כללי שייט בכל פעם שה ADT של הרציף מתפנה. לייתר דיוק, אם ה ADT פנוי לחלוטין (קרי, לא נמצא בתוכו אף כלי שייט) ומספר כללי השייט הממתינים ב Barrier הוא לכל הפחות M, אזי M כללי שייט, מראש תור ההמתנה של ה Barrier, ייקבלו Signal ויכנסו ל ADT של הרציף (שלב מס' 2 בתרשים לעיל – בדוגמא זו M הוא, כאמור, 2).

5.7.2. ברגע שכלי שייט כלשהו נכנס ל ADT של הרציף, הוא:

- i. מדפיס אינדיקציה על כניסתו לרציף הפריקה (בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי השייט והשלב בו הוא נמצא – פירוט נוסף לגבי ההדפסות יינתן בפרק 6);
- ii. מבצע הגרלה של זמן שינה (בטווח מסויים שייפורט בפרק 6) ולבצע Sleep לפרק זמן זה, ע"מ לדמות שתהליך הכניסה לרציף לוקח זמן;
- iii. מתמקם מול ה Crane הפנוי הראשון, ע"מ שזה (המנוף/עגורן) ייפרוק ממנו את המטען שלו (שלב מס' 3 בתרשים לעיל). כאמור בסעיף 1.7, הקונספט הנ"ל מבטיח שלכל כלי שייט יהיה מנוף שמוקדש לו, לצורך הפריקה.
- iv. מדפיס אינדיקציה לגבי איזה Crane משרת אותו (בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי השייט והשלב בו הוא נמצא – פירוט נוסף לגבי ההדפסות יינתן בפרק 6).

5.7.3. תהליך הפריקה הוא כדלהלן:

- i. על כלי השייט להגריל מספר בין 5 ל 50, כאשר מספר זה מהווה את משקל המטען שלו (בטונות);

ii. לאחר (i), על כלי השייט להדפיס את משקל המטען שלו בטונות (בכל הדפסה

כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי השייט והשלב בו הוא נמצא – פירוט נוסף לגבי ההדפסות יינתן בפרק 6). לאחר מכן, עליו להעביר את המספר הזה ל Crane, מולו התמקם, ולשחרר אותו לעבודת הפריקה (שימו לב לנאמר בסעיף 5.6.4). כל עוד ה Crane לא סיים את עבודתו, אסור לכלי השייט להמשיך לשלב הבא ב ADT של הרציף;

iii. כאשר ה Crane נכנס לעבודה, עליו לבצע הגרלה של זמן שינה (בטווח מסויים שייפורט בפרק 6) ולבצע Sleep לפרק זמן זה, ע"מ לדמות שתהליך הפריקה לוקח זמן. לאחר מכן, עליו להדפיס את משקל המטען שפרק (בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של ה Crane והשלב בו הוא נמצא – פירוט נוסף לגבי ההדפסות יינתן בפרק 6); ומיד אח"כ, לתת אינדיקציה לכלי השייט הנפרק, שתהליך הפריקה הסתיים;

iv. לאחר שכלי השייט מקבל אינדיקציה מהמנוף/עגורן (מולו התמקם) על סיום תהליך הפריקה, הוא ממשיך לשלב היציאה מן ה ADT של רציף הפריקה (שלב מס' 4 בתרשים לעיל).

5.7.4. שלב היציאה של כלי שייט מן ה ADT של רציף הפריקה (שלב מס' 4 בתרשים לעיל):

- i. על כלי השייט לבצע הגרלה של זמן שינה (בטווח מסויים שייפורט בפרק 6) ולבצע Sleep לפרק זמן זה, ע"מ לדמות שתהליך היציאה מן הרציף לוקח זמן;
- ii. על כלי השייט להדפיס אינדיקציה על יציאתו מרציף הפריקה (בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי השייט והשלב בו הוא נמצא – פירוט נוסף לגבי ההדפסות יינתן בפרק 6);
- iii. מבצע יציאה מן ה ADT של רציף הפריקה ומשם ממשיך עפ"י האמור בסעיף 4.3.

5.8. Barrier:

5.8.1. הקונספט של Barrier קלאסי, שתואר בתחילת פרק 5, מתאר Barrier, המהווה

נקודת המתנה וסינכרון לכל ה Threads במערכת: כל ה Threads במערכת

משתחררים יחדיו (בו-זמנית) ממצב המתנה, כאשר כולם הגיעו ל Barrier. ה Barrier של רציף הפריקה, אמנם מבוסס על קונספט זה, אך שונה באופן מהותי ממנו: כפי שתואר בסעיף 5.7.1, ה Barrier של רציף הפריקה אחראי להזין את ה ADT של הרציף, ע"י יצירת וויסות מדורג של הכנסת כללי שייט, ל ADT של רציף הפריקה, בקבוצות של M כללי שייט בכל פעם (כאשר M הוא מספר המנופים ברציף). בהיבט הזה, ה Barrier של רציף הפריקה – גם הוא סוג של ADT.

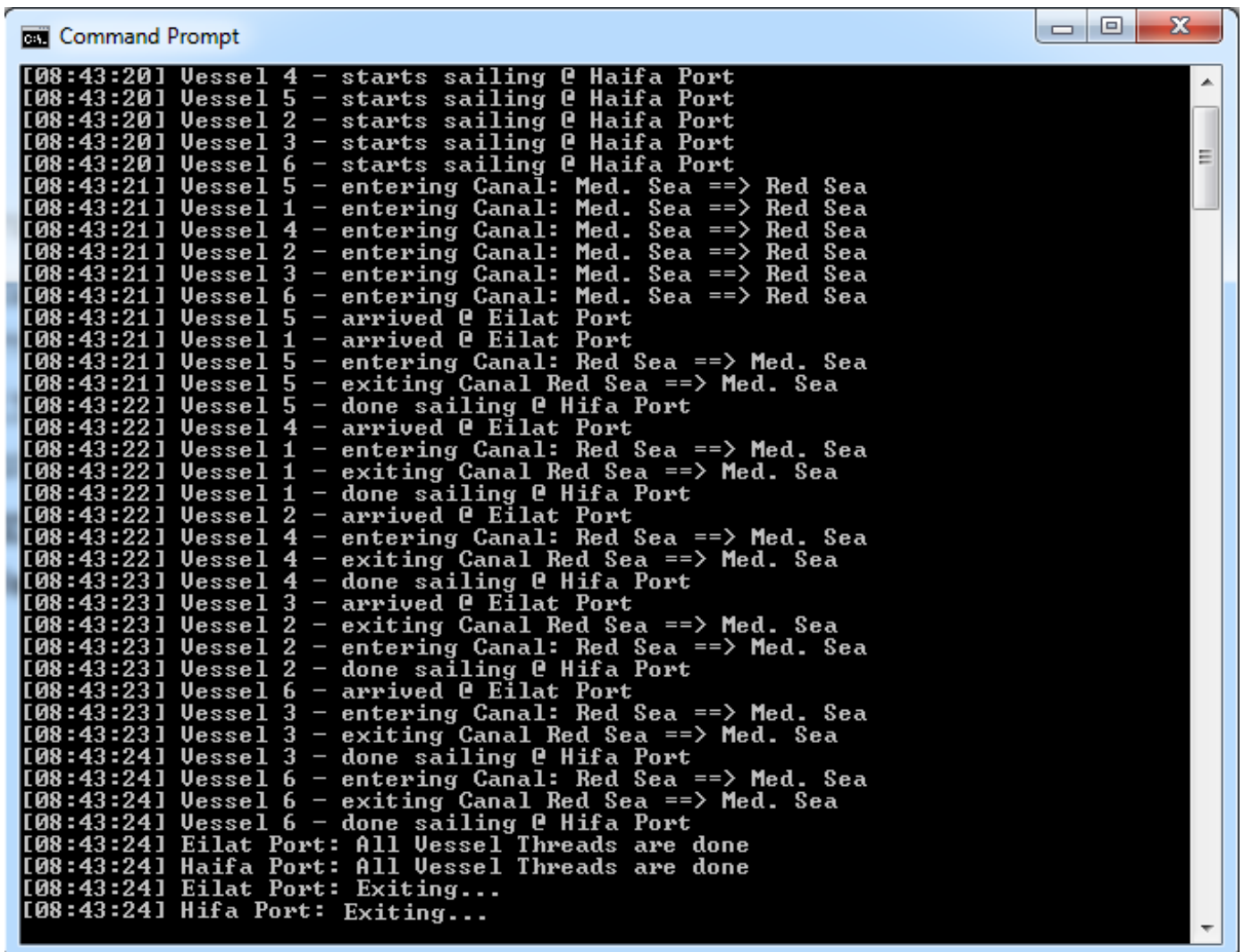
5.8.2. עליכם לתכנן הייטב את מרכיבי ה Barrier ולממשם, עפ"י העקרונות הבאים:

- i. ל Barrier יש תור המתנה, המנוהל בצורת FIFO (ראו בתרשים לעיל) כאשר, כללי שייט שמגיעים ל Barrier ממתינים בו (נכנסים למצב WAIT), עד שמגיע תורם להכנס ל ADT של רציף הפריקה;
- ii. אם ה ADT של רציף הפריקה, פנוי לחלוטין (קרי, לא נמצא בתוכו אף כלי שייט) ומספר כללי השייט הממתינים ב Barrier הוא לכל הפחות M, אזי M כללי שייט, מראש תור ההמתנה של ה Barrier, ייקבלו Signal וייכנסו ל ADT של הרציף (שלב מס' 2 בתרשים לעיל – בדוגמא זו M הוא, כאמור, 2);
- iii. בהמשך ל (ii), נחدد את הנקודה: בנקודת זמן בה שני התנאים, המתוארים ב (ii), מתקיימים (כלומר, רציף הפריקה, פנוי לחלוטין ומספר כללי השייט הממתינים ב Barrier הוא לכל הפחות M), M כללי השייט הראשונים שהגיעו ל Barrier חייבים להשתחרר מיד ולהמשיך (לשלב 2 בתרשים לעיל) – **אסור בתכלית האיסור לעכב אותם ברגע שהמצב הזה מתקיים!**
- iv. בכל סיטואציה אחרת, השוללת את קיומם של שני התנאים ב (ii), אסור לשחרר אף כללי שייט מה Barrier;
- v. בהמשך לסעיף 4.3, כאשר כלי שייט מגיע לרציף הפריקה (Unloading Quay) הדבר הראשון שהוא מבצע זה, למעשה, כניסה ל Barrier (שלב מס' 1 בתרשים לעיל) – מיד עם כניסתו ל Barrier, על כלי השייט להדפיס אינדיקציה על כך (בכל הדפסה כזו יופיעו פרטים על: נק' הזמן, המזהה הייחודי של כלי השייט והשלב בו הוא נמצא – פירוט נוסף לגבי ההדפסות יינתן בפרק 6).

6. דגשים I:

6.1. הדפסות: בסעיף זה נתייחס להדפסות היוצאות מהסימולציה ל console (מסך ההרצה)

בשלבם השונים, ע"י ה Entities השונים. העיקרון המנחה הוא לבצע הדפסות בנק' חשובות, שייתרמו להבנה ויכולת ניתוח טובה של מי שמפעיל את הסימולציה. באופן כללי, פורמט ההדפסות הוא כדלהלן: בכל הדפסה, יופיעו פרטים על נק' הזמן (בפורמט – [HH:MM:SS]), המזהה הייחודי של ה Thread (במידה ולא מדובר ב main Threads של הנמלים) ופירוט תמציתי לגבי השלב בו הוא נמצא. להלן, דוגמא להדפסות הרצת הסימולציה – שימו לב שדוגמא זו אינה כוללת את ההדפסות הנדרשות מתוך השלבים השונים ברציף הפריקה (Unloading Quay) של נמל אילת; כמו כן, אינה כוללת הודעות נוספות שנדרשות מה main Threads:



```
[08:43:20] Vessel 4 - starts sailing @ Haifa Port
[08:43:20] Vessel 5 - starts sailing @ Haifa Port
[08:43:20] Vessel 2 - starts sailing @ Haifa Port
[08:43:20] Vessel 3 - starts sailing @ Haifa Port
[08:43:20] Vessel 6 - starts sailing @ Haifa Port
[08:43:21] Vessel 5 - entering Canal: Med. Sea ==> Red Sea
[08:43:21] Vessel 1 - entering Canal: Med. Sea ==> Red Sea
[08:43:21] Vessel 4 - entering Canal: Med. Sea ==> Red Sea
[08:43:21] Vessel 2 - entering Canal: Med. Sea ==> Red Sea
[08:43:21] Vessel 3 - entering Canal: Med. Sea ==> Red Sea
[08:43:21] Vessel 6 - entering Canal: Med. Sea ==> Red Sea
[08:43:21] Vessel 5 - arrived @ Eilat Port
[08:43:21] Vessel 1 - arrived @ Eilat Port
[08:43:21] Vessel 5 - entering Canal: Red Sea ==> Med. Sea
[08:43:21] Vessel 5 - exiting Canal Red Sea ==> Med. Sea
[08:43:22] Vessel 5 - done sailing @ Hifa Port
[08:43:22] Vessel 4 - arrived @ Eilat Port
[08:43:22] Vessel 1 - entering Canal: Red Sea ==> Med. Sea
[08:43:22] Vessel 1 - exiting Canal Red Sea ==> Med. Sea
[08:43:22] Vessel 1 - done sailing @ Hifa Port
[08:43:22] Vessel 2 - arrived @ Eilat Port
[08:43:22] Vessel 4 - entering Canal: Red Sea ==> Med. Sea
[08:43:22] Vessel 4 - exiting Canal Red Sea ==> Med. Sea
[08:43:23] Vessel 4 - done sailing @ Hifa Port
[08:43:23] Vessel 3 - arrived @ Eilat Port
[08:43:23] Vessel 2 - exiting Canal Red Sea ==> Med. Sea
[08:43:23] Vessel 2 - entering Canal: Red Sea ==> Med. Sea
[08:43:23] Vessel 2 - done sailing @ Hifa Port
[08:43:23] Vessel 6 - arrived @ Eilat Port
[08:43:23] Vessel 3 - entering Canal: Red Sea ==> Med. Sea
[08:43:23] Vessel 3 - exiting Canal Red Sea ==> Med. Sea
[08:43:24] Vessel 3 - done sailing @ Hifa Port
[08:43:24] Vessel 6 - entering Canal: Red Sea ==> Med. Sea
[08:43:24] Vessel 6 - exiting Canal Red Sea ==> Med. Sea
[08:43:24] Vessel 6 - done sailing @ Hifa Port
[08:43:24] Eilat Port: All Vessel Threads are done
[08:43:24] Haifa Port: All Vessel Threads are done
[08:43:24] Eilat Port: Exiting...
[08:43:24] Hifa Port: Exiting...
```


6.1.1. הדפסות כלי השייט (Vessel):

- i. אינדיקציה על תחילת השייט;
- ii. אינדיקציה על מעבר בנתיב התעלה בכוון מהים התיכון לים האדום;
- iii. אינדיקציה על הגעה לנמל אילת;
- iv. כל האינדיקציות לגבי השלבים השונים ברציף הפריקה (Unloading Quay) של נמל אילת (הדברים מפורטים בפרק 5);
- v. אינדיקציה על מעבר בנתיב התעלה בכוון מהים האדום לים התיכון;
- vi. אינדיקציה על חזרה לנמל חיפה;
- vii. אינדיקציה על סיום השייט.

6.1.2. הדפסות מנוף/עגורן (Crane):

- i. אינדיקציה על יצירתו;
- ii. כל האינדיקציות לגבי השלבים השונים ברציף הפריקה (Unloading Quay) של נמל אילת (הדברים מפורטים בפרק 5);
- iii. אינדיקציה על סיום תפקידו (קרי, סיום חייו).

6.1.3. הדפסות ה main Thread של נמל חיפה:

- i. אינדיקציה על מספר כללי השייט;
- ii. אינדיקציה על בקשת אישור מעבר מנמל אילת;
- iii. אינדיקציה על תשובת נמל אילת (חיובית/שלילית), כתגובה ל (ii);
- iv. אינדיקציה על סיום כל כללי השייט;
- v. אינדיקציה על סיום ה (Exit) Process של נמל חיפה.

6.1.4. הדפסות ה main Thread של נמל אילת:

- i. אינדיקציה על בקשת אישור מעבר מנמל חיפה ומספר כללי השייט הצפוי;
- ii. אינדיקציה על התשובה הנשלחת לנמל חיפה (חיובית/שלילית), כתגובה ל (i);
- iii. אינדיקציה על סיום כל כללי השייט;
- iv. אינדיקציה על סיום כל המנופים/עגורנים (Cranes);
- v. אינדיקציה על סיום ה (Exit) Process של נמל אילת.

6.1.5. הדפסות נוספות: יש לזכור לתמוך בהדפסות משמעותיות במקרים חריגים ברורים (כגון: כישלון הקצאות משאבים, המקרים המתוארים בפרקים 3 ו 4, ועוד).

6.2. **Sleep**: במקרים מסויימים, על ה Threads במערכת לבצע הגרלה של זמן שינה (בטווח מסויים שייפורט מייד) ולבצע Sleep לפרק זמן זה, ע"מ: להאט מעט את קצב הדברים (בכדי שתוכלו לעקוב אחריהם), לדמות שתהליכים מסויימים לוקחים זמן, להגביר את המחלוקת/תחרות (contention) במקרים מסויימים. העיקרון המנחה הוא לבצע Sleep בנק' חשובות, כאשר זמן השינה, בכל פעם, צריך להיות ערך מוגרל בטווח הערכים שבין 5 milliseconds ל 3 seconds.

6.2.1 Sleep בכלי שייט (Vessel):

- i. מייד לאחר אינדיקציית ההדפסה שלו על תחילת השייט (6.1.1, נק' (i));
- ii. מייד לאחר אינדיקציית ההדפסה שלו על המעבר בתעלה בכוון הים האדום (6.1.1, נק' (ii)) – וזאת לפני עזיבת התעלה;
- iii. מייד לאחר אינדיקציית ההדפסה שלו על הגעתו לנמל אילת (6.1.1, נק' (iii));
- iv. כל ה Sleeps בשלבים הרלוונטיים ברציף הפריקה (Unloading Quay) של נמל אילת (הדברים מפורטים בפרק 5);
- v. מייד לאחר אינדיקציית ההדפסה שלו על המעבר בתעלה בכוון הים התיכון (6.1.1, נק' (v)) – וזאת לפני עזיבת התעלה;
- vi. מייד לאחר לאחר אינדיקציית ההדפסה שלו על הגעתו לנמל חיפה (6.1.1, נק' (vi)).

6.2.2 Sleep במנוף/עגורן (Crane): כל ה Sleeps בשלבים הרלוונטיים ברציף הפריקה (Unloading Quay) של נמל אילת (הדברים מפורטים בפרק 5).

7. דגשים II:

7.1. על המימוש להיות מודולרי, גנרי ואלגנטי וניתן לשינויים והרחבות בקלות.

7.2. על המימוש להיות יעיל הן מבחינת צריכת המשאבים, והן מבחינת האיזון הראוי בין צריכת משאבים לבין Synchronization, עפ"י העקרונות שנלמדו בקורס. * זיכרו: סינכרון ייתר (Over-Synchronization) יכול לגרום לתופעות כגון: בזבז משאבים, עיכובים שלא לצורך, Deadlock; לעומת זאת, סינכרון בחסר (Synchronization Deficiency) יכול לגרום לתופעות כגון: Race Conditions, Loss-Updates.

7.3. כחלק מיעילות צריכת המשאבים, על המימוש לדאוג לשחרור משאבים מייד כשלא צריך אותם – אסור לעכב שחרור משאב, מהרגע שאין בו צורך יותר.

8. דגשים III:

8.1. יש להגיש את התוצר המלא הסופי, לאחר ביצוע כל הדרישות (החלוקה לסעיפים נועדה עבורכם – לצורך הדרגתיות ההבנה, התוספות והשינויים, ע"מ שתוכלו לבדוק היטב כל תוספת/שינוי, לפני שאתם ממשיכים הלאה). למען הסר ספק, בדומה לכל מטלה אחרת שהגשתם בקורס, פרוייקט המקור שהגשתם בכל אחת מתתי הספריות: HaifaSrc ו EilatSrc של הגשת המטלה המסכמת חייב להתקמפל ולרוץ אצלי במחשב, ללא שום תיקונים או תלויות/התאמות שעלי לבצע! לפיכך:

8.1.1. חובה לכתוב את המטלה בסביבת VS2019 בלבד!

8.1.2. שימוש ב Event Objects של Win32 API אסור בתכלית האיסור!

8.1.3. אם השתמשתם בקבצי מקור נוספים (כגון: קבצי h. ו/או קבצי c.), עליכם לצרפם לספריית ה Src הרלוונטית (מבין השתיים: HaifaSrc ו EilatSrc), יחד עם קובץ המקור העיקרי לפרוייקט הרלוונטי;

8.1.4. אסור שתהיינה תלויות במסלול (path) כזה או אחר של מיקום הקבצים. כלומר:

- i. שני הפרוייקטים חייבים להתקמפל ברגע שיוצרים עבור כל אחד מהם פרוייקט VS20019 חדש ומייבאים לתוכו את כל קבצי המקור מתוך ספריית Src הרלוונטית (מבין השתיים: HaifaSrc ו EilatSrc);
- ii. הפרוייקט בכלותו חייב לרוץ מתוך הרצת קובץ ה .exe של "נמל חיפה", באמצעות חלון CMD, וזאת ללא קשר להיכן הוא ממוקם. ניתן להניח כי שני קבצי ה .exe של שני הפרוייקטים ("נמל חיפה" ו "נמל אילת") נמצאים באותו ה path. למעשה, אם תיצרו את ה Process של "נמל אילת" מתוך ה Process של "נמל חיפה" עפ"י הנחה זו, תיצרו באופן אוטומטי חוסר תלות במיקומים ספציפיים של קבצי ה .exe.

8.1.5. אני ממליץ לכם בחום לבדוק את עצמכם לפני ההגשה - כלומר, להעתיק את שני קבצי ה .exe של שני הפרוייקטים ("נמל חיפה" ו "נמל אילת") לספריה כלשהי במחשב של חבר/ה אחר/ת בקבוצה, לפתוח לשם חלון CMD ולהריץ את הסימולציה משורת הפקודה במיקום זה – וזאת ע"מ לוודא את עצמכם. **זיכרו: פרוייקט שלא ייתקמפל ו/או לא ירוץ - לא ייבדק!**

Good Luck!