

# עבודה 3 - עקרונות תכנות מונחה עצמים 2017

מתרגל אחראי: הודי גולדמן.

## 1 הסבר כללי:

במשימה זו תצטרכו לממש את המשחק **סוקובאן**. כולל ממשק משתמש גרפי שימומש ב-SWING. התוכנית נבדקת על פי נכונות המשחק, ציות לעקרונות תכנות מונחה עצמים כפי שנלמדו בכיתה, מידול המערכת (ע"י דיאגרמת מחלקות), שימושיות משתמש ואטראקטיביות. אתם מוזמנים להוסיף תוספות (פיצ'רים) משלכם ואף לזכות בנקודות בונוס (פירוט בהמשך). סך הציון בתרגיל יכול להגיע ל 120, כאשר מתוכן 20 נקודות הן הנקודות של הבונוס. עליכם להתחיל את המטלה בקריאת של מסמך זה עד הסוף, לאחר מכן יש תייצר תרשים מחלקות (UML) המתאים למערכת, ולבסוף יש לממש את המשחק על פי הדרישות. אם אתם לא מכירים את המשחק **סוקובאן**, ניתן לקרוא עליו [כאן](#), ולשחק בו אונליין [כאן](#). בהמשך מפורטים חוקי המשחק למי שאינו מכיר את המשחק.

## 2 חוקי המשחק:

המשחק סוקובאן (מנהל מחסן) הוא משחק לוח לשחקן יחיד. הלוח הוא לוח תאים מרובע (TILE), כל תא (מיקום) בלוח יכול להיות חומה (חסום) או ריצפה, בנוסף תא ריצפה יכול להכיל אחד מהבאים:

1. דמות של השחקן ("מחסנאי") שכל רגע נתון נמצא על אחד מתאי הריצפה, ובכל שלב יכול לבצע צעד לאחד מתאי הריצפה הסמוכים אליו מימינו, משמאלו, מעליו או מתחתיו, ע"פ החוקים שיפורטו בהמשך.
2. ארגזים – במשחק יש מספר ארגזים, כל תא ריצפה יכול להכיל לכל היותר ארגז אחד.
3. תאי איכסון (תאי יעד) – יש במשחק תאי איכסון כמספר הארגזים במשחק, תא איכסון הוא בעצם תא ריצפה מיוחד.

יש שלבים (לוחות משחק) שונים למשחק זה כאשר לכל לוח יש גודל משלו (אורך ורוחב), סידור הלוח (איזה תא הוא חומה ואיזה ריצפה), מיקום התחלתי של ה"מחסנאי" ושל הקופסאות, ומיקום תאי האיכסון.

### 2.1 המטרה:

מטרת השחקן היא "לדחוף" כל ארגז לתא איכסון כלשהו, המשחק נגמר (בניצחון) כאשר כל ארגז נמצא בתא איכסון.

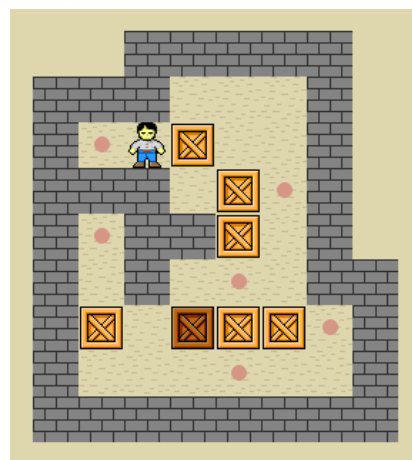
## 2.2 מהלך המשחק:

בכל צעד במשחק ניתן להזיז את המחסנאי תא אחד (ימינה, שמאלה, למעלה או למטה) בעזרת מקשי החצים במקלדת, מתא ריצפה אחד לתא ריצפה אחר, כלומר לא ניתן לעבור לתא שהוא חומה. בנוסף אם עוברים לתא שמכיל ארגז אז השחקן "דוחף" את הארגז לתא הבא באותו כיוון, אך לא ניתן לדחוף יותר מארגז אחד בו זמנית ולכן אם התא שאליו עוברים מכיל ארגז, אז התא הבא באותו כיוון חייב להיות ריק, כדי שהארגז יעבור אליו (ראו דוגמאות מטה).

## 2.3 דוגמאות:

מצב לפני לחיצה על חץ שמאלה ( ← )	מצב אחרי הלחיצה
	<b>צעד אסור - אין שינוי</b> לא ניתן לדחוף יותר מארגז אחד
	<b>צעד אסור - אין שינוי</b>
	צעד רגיל, ללא דחיפת ארגז
	צעד של דחיפת ארגז

דוגמת ריצה של משחק (מתוך [ויקיפדיה](#)):



תאי היעד מסומנים בעיגול אדום

### 3 דרישות מימוש:

עליכם לממש את המשחק במלואו בשפת JAVA ובעזרת ממשק המשתמש הגרפי SWING.

#### 3.1 דיאגרמת מחלקות:

יש לעצב את המערכת בהתאם לעקרונות תכנות מונחה עצמים כפי שנלמד בכיתה. לפני תחילת כתיבת הקוד עליכם לתכנן ולעצב את המערכת ולשקף זאת ע"י דיאגרמת מחלקות (UML) שמתארת את המחלקות העיקריות במערכת ואת הקשרים ביניהן.

#### 3.2 טעינת שלבים:

במשחק זה יש מספר שלבים, כל השלבים שמורים בקובץ טקסט נפרד ומסודרים לפי הסדר, כלומר שלב ראשון יופיע ראשון בקובץ (ראו קובץ levels.txt המצורף). המשתמש יכול לבחור באיזה שלב לשחק והמצב ההתחלתי של שלב זה "יטען" מתוך הקובץ. מצורף מימוש של המחלקה LevelLoader המאפשרת לקרוא את השלבים מקובץ ולייצר מופע של לוח השלב המתאים. המחלקה הנ"ל מייצרת מטריצה (מערך דו מימדי) של תאים (Cell) המיצג מצב התחלתי של השלב המבוקש, המחלקה Cell המייצגת תא בלוח המשחק. עליכם להשתמש במחלקה LevelLoader לטעינת לוח השלב המתאים בהתאם לבחירת המשתמש. בהפעלת המשחק יופיע כברירת מחדל השלב הראשון. ניתן להוסיף להרחיב (להוסיף שיטות או שדות) בהתאם לצורך, ומומלץ להוסיף שלבים נוספים לקובץ השלבים כדי לבדוק את התנהגות המערכת במצבים שונים.

#### 3.3 ממשק משתמש גרפי:

הלוח הראשי (המחלקה Board):

- יש להציג בצורה נוחה וברורה את מצב הלוח בכל רגע נתון, הלוח יהיה מורכב מ 6 סוגים שונים של תאים, כלומר שיהיה אפשר להבדיל בניהם באופן ויזואלי:
  - תא קיר
  - תא ריצפה ריקה
  - תא ריצפה עם המחסנאי
  - תא ריצפה עם ארגז
  - תא איכסון
  - תא איכסון עם ארגז

#### 3.4 חלון המשחק:

בכל זמן המשחק בנוסף ללוח הראשי יופיע:

- הצגת השלב הנוכחי (בהפעלת המשחק יופיע שלב 1)
- אפשרות לעבור לשלב אחר מתוך כל השלבים הקיימים
- הצגה של מספר הצעדים שבוצעו מתחילת המשחק עד כה (ניקוד)
- אפשרות להתחלת המשחק מחדש, בעת בחירה באפשרות זו, לוח המשחק יחזור למצב ההתחלתי והמונה של מספר הצעדים יתאפס

- בונוס (10 נק'): אפשרות של UNDU לביטול הפעולה האחרונה כדי לממש יכולת זו יש לממש מנגנון של מחסנית פעולות שתשמור את כל הפעולות שבוצעו מתחילת המשחק, ויש לאפשר לבצע פעולות UNDU בכל רגע עד שמגיעים חזרה למצב ההתחלתי.

### 3.5 ניצחון:

ברגע שכל הארגזים נמצאים בתאי אכסון יש לדווח למשתמש על ניצחון במשחק. פרט לאמור לעיל אתם חופשיים לעצב את המשחק לפי בחירתכם ולהוסיף פיצ'רים ויזואליים או קוליים. יינתנו עד 10 נקודות בונוס על עיצוב מקורי ו\או על שימושיות ואטרקטיביות של המשחק (על פי שיקול הבודק).

## 4 הוראות הגשה:

יש להגיש דרך ה system submission קובץ zip יחיד בשם hw3.zip שמכיל:

1. מסמך pdf בשם hw3.pdf עם תרשים מחלקות של המערכת
2. קובץ jar בשם hw3.jar שמכיל את קבצי המקור, הקבצים המקומפלים וכל משאב אחר בו השתמשתם (לדוגמא קבצי תמונות)
1. אנא בדקו שאתם מצליחים להריץ את קובץ ה jar שלכן מהקומנד ליין (CMD).
2. אנא ודאו שקוד המקור מופיע בתוך קובץ ה-jar (פתחו אותו עם תוכנת דחיסה, למשל WINZIP, ותוודאו שקבצי המקור נמצאים).
3. קובץ השלבים

## 5 מפתח ניקוד:

1. תרשים מחלקות – 10%
2. נכונות (התוכנה עומדת בכל חוקי המשחק) – 10%
3. קוד(מעוצב נכון, לוגיקה נכונה, מסודר וכתוב על פי קונבנציות) – 40%
4. ממשק גרפי (אטרקטיבי, שימושי, מתמודד יפה עם לוחות בגדלים שונים) – 40%
5. בונוס 1 – אפשרות לביטול המהלך האחרון – 10%
6. נקודות נוספות ינתנו ע"פ שיקול הבודק (לכל היותר 10 נקודות) על:
  1. ממשק גרפי מרשים במיוחד
  2. פיצ'רים מקוריים שאינם מופיעים בדרישות.

**בהצלחה!**