

מכללת הדסה, החוג למדעי המחשב  
תכנות מונחה עצמים ופיתוח משחקים  
סמסטר ב', תשפ"ד

## תרגיל 3

תאריך אחרון להגשה:

הנביאים - יום ה', 30/05/23, כ"ב אייר ה'תשפ"ד, בשעה 23:59

שטראוס-גברים - ?

### מטרת התרגיל:

שימוש במבני נתונים, איטרטורים ואלגוריתמים בסגנון STL וריענון השימוש ב-SFML.

### תיאור כללי:

בתרגיל זה נבנה משחק פשוט של דוקים תוך שימוש ב-SFML.

דוגמא אפשרית למשחק: <http://www.kongregate.com/games/cranberrygame/pick-up-sticks>

### כללי המשחק:

כללי המשחק פשוטים מאוד: לשחקן מוצג לוח שעליו מפוזרים דוקים (מקלות צבעוניים) ועליו להרים אותם בזה אחר זה. אפשר להרים דוק רק אם אין מעליו דוקים אחרים. אם השחקן ינסה להרים דוק שאסור להרים, הפעולה לא תתבצע.

אלגוריתם לבדיקה האם שני קטעים נחתכים - [בדיקת חיתוך קטעים](#)

המשחק מוגבל בזמן (הזמן הראוי נתון לבחירתכם; אולי תלוי במספר הדוקים שתשימו על הלוח).

המשחק מסתיים כשהלוח מתרוקן או כשהזמן חלף.

כל דוק מזכה את השחקן במספר נקודות שונה (לבחירתכם אם הניקוד נקבע אקראית ביצירת הדוק או תלוי בצבע).

## הלוח:

מספר הדוקים בלוח מוגרל אקראית מטווח שתגדירו (ניתן לממש רמות שונות של המשחק ולהגדיר טווח שונה לפי רמה).

פיזור הדוקים בתחילת משחק מתבצע באופן אקראי. לכל דוק נגריל ערכים עבור נקודת התחלה, אורך, זווית שבה נסובב אותו וצבע. כאשר השחקן ילחץ על דוק שניתן להרים, הדוק יעלם. כאשר השחקן ילחץ על דוק שלא ניתן להרים, תהיה **אינדקציה** למה לא ניתן להרים דוק זה. באפשרותכם לבחור איך לממש אינדקציה זו: ניתן לסמן רק את הדוק שנמצא ישירות מעליו וניתן לסמן את כל הדוקים שנמצאים מעליו. "סימון" יכול להיות "היבהוב" בצבע מסוים (נניח שחור) לזמן קצר או יצירת מסגרת בצבע צהוב זרחני לזמן קצר.

שימו לב לצייר את הדוקים בסדר הנכון, כך שהדוק העליון אכן יצויר מעל כל השאר.

בנוסף יוצגו בלוח הנתונים הבאים:

- כמה זמן נותר
- כמה דוקים נשארו
- כמה דוקים הורמו
- כמה דוקים ניתן להרים כרגע

בנוסף, יהיה כפתור **רמז**. כפתור זה עוזר לשחקן למצוא את הדוקים אותם ניתן להרים. כאשר השחקן ילחץ על כפתור זה, הדוקים שניתן כרגע להרים יבהבו בדומה **לאינדקציה** כאשר לא ניתן להרים דוק. סדר הצביעה צריך להיות לפי מספר הנקודות של כל דוק. דוק עם הניקוד הגבוה ביותר יצבע ראשון ודוק עם הניקוד הנמוך ביותר יצבע אחרון. שימו לב שצריכה להיות השהייה בין צביעה של שני דוקים.

## שמירה וטעינת משחק:

לשחקן תהיה אפשרות לשמור משחק ולטעון משחק שמור ולהמשיך באותה נקודה. השמירה של המשחק תהיה לתוך קובץ. פורמט שמירת הנתונים נתון להחלטתכם, רק חשוב שהמשחק ימשיך מאותה נקודה – כלומר שהמצב של הלוח ושל הזמן שנותר יישארו זהים כפי שהיו בעת השמירה. שימו לב שאתם צריכים להתמודד עם מצב שבו הקובץ שהשחקן בוחר על מנת לטעון משחק אינו מתאים לפורמט ולהוציא הודעת שגיאה בהתאם. אתם מתבקשים לממש זאת באמצעות זריקת חריגות. צריכות להיות 2 סוגי חריגות לפחות:

1. חריגה של קובץ לא תקין – קובץ לא קיים, פורמט לא מתאים, אותיות במקום מספרים וכו'.
2. חריגה של נתון לא הגיוני – לדוגמה הזמן שנותר למשחק הוא יותר מהזמן המקסימלי האפשרי.

## מסכים נוספים:

מלבד המשחק עצמו, ניצור מסך התחלה עם תפריט בסיסי. התפריט יכיל לפחות את האפשרויות של התחלת משחק ושל יציאה. כמו כן, נציג מסך סיום שיציג את תוצאת המשחק.

## הנחיות למימוש:

- הדוק לא מונח במקביל לציר ה-Y (אחרת זה מסבך את אלגוריתם בדיקת החיתוך. ראו את אלגוריתם בדיקת החיתוך [בקישור](#))
- מבני הנתונים שתדרשו להשתמש בהם הם מבני נתונים של STL
- נחלק את המשחק לשני חלקים:
  - a. **פיזור הדוקים ובניית מבני הנתונים** – בחלק זה אנחנו יכולים לסבול עיכוב ולכן לא חייבים להיות סופר יעילים מבחינת זמן ריצה. חישבו איך לממש את החלק הזה כדי שבחלק של המשחק עצמו תצליחו לעמוד בהנחיות.
  - b. **המשחק עצמו** – במשחק עצמו נרצה שהפעולות תהיינה יעילות ככל הניתן לפי ההוראות בהמשך.
- הנחיות זמן ריצה ומקום:
  - a. מציאת הדוק אחרי לחיצה על העכבר – זמן ריצה  $O(N)$ , כאשר N הוא מספר הדוקים שכרגע על המסך.
  - b. בדיקה האם ניתן להרים דוק מסוים – פעולה זו צריכה להתבצע **בזמן קבוע**.
  - c. כמות הדוקים שצריכה להיות בזיכרון תהיה לפי מספר הדוקים שנשארו – נניח שיש לכם class Stick, אז מספר האובייקטים מסוג המחלקה הזו תמיד יהיה מספר הדוקים שנשארו. יכולים להיות מספר מצביעים לאותו אובייקט.
  - d. זמן הריצה של ציור הדוקים שניתן להרים בעקבות לחיצה על כפתור **רמז** יהיה ליניארי בהתאם לכמות הדוקים שניתן להרים – אם מספר הדוקים שניתן להרים הוא K אז זמן הריצה צריך להיות  $O(K)$ .

## קובץ ה-README

יש לכלול קובץ README שיקרא README.doc, README.docx או README.txt (ולא בשם אחר). הקובץ יכול להיכתב בעברית ובלבד שיכיל את הסעיפים הנדרשים.

קובץ זה יכיל לכל הפחות:

1. כותרת.
  2. פרטי הסטודנט: שם מלא כפי שהוא מופיע ברשימות המכללה, ת"ז.
  3. הסבר כללי של התרגיל.
  4. רשימה של הקבצים שיצרנו, עם הסבר קצר (לרוב לא יותר משורה או שתיים) לגבי תפקיד הקובץ.
  5. מבני נתונים עיקריים ותפקידיהם.
  6. אלגוריתמים הראויים לציון.
  7. **תיכון (design): הסבר קצר מהם האובייקטים השונים בתוכנית, מה התפקיד של כל אחד מהם וחלוקת האחריות ביניהם ואיך מתבצעת האינטראקציה בין האובייקטים השונים.**
  8. באגים ידועים.
  9. הערות אחרות.
- יש לתמצת ככל שניתן אך לא לוותר על אף חלק. אם אין מה להגיד בנושא מסוים יש להשאיר את הכותרת ומתחתיו פסקה ריקה. **נכתוב ב-README כל דבר שרצוי שהבודק ידע כשהוא בודק את התרגיל.**

## אופן ההגשה

הקובץ להגשה: ניתן ליצור בקלות קובץ zip המותאם להגדרות ההגשה המפורטות להלן ישירות מתוך VS, כפי המוסבר תחת הכותרת "יצירת קובץ ZIP להגשה או לגיבוי" בקובץ "הנחיות לשימוש ב-Visual Studio 2022". אנא השתמשו בדרך זו (אחרי שהגדרתם כראוי את שמות הצוות ב-MY\_AUTHORS: **נשים את שמות המגישים בתוך המרכאות, נקייד להפריד בין השם הפרטי ושם המשפחה בעזרת קו תחתי ואם יש יותר ממגיש אחד, נפריד בין השמות השונים בעזרת מקף (מינוס '-')**) וכך תקבלו אוטומטית קובץ zip המותאם להוראות, בלי טעויות שיגררו אחר כך בעיות בבדיקה.

באופן כללי, הדרישה היא ליצור קובץ zip בשם oop2\_exN-firstname\_lastname.zip (או במקרה של הגשה בזוג – oop2\_exN-firstname1\_lastname1-firstname2\_lastname2.zip), כשהקובץ כולל את כל קובצי הפרויקט, למעט תיקיות out ו-vs. כל הקבצים יהיו בתוך תיקייה ראשית אחת.

את הקובץ יש להעלות ל-Moodle של הקורס למשימה המתאימה. בכל מקרה, **רק אחד** מהמגישים יגיש את הקובץ ולא שניהם.

**הגשה חוזרת:** אם מסיבה כלשהי החלטתם להגיש הגשה חוזרת יש לוודא ששם הקובץ זהה לחלוטין לשם הקובץ המקורי. אחרת, אין הבודק אחראי לבדוק את הקובץ האחרון שיוגש.

כל שינוי ממה שמוגדר פה לגבי צורת ההגשה ומבנה ה-README עלול לגרום הורדת נקודות בציון.

מספר הערות:

1. נשים לב לשם הקובץ שאכן יכלול את שמות המגישים.

2. נשים לב לשלוח את תיקיית הפרויקט כולה, לא רק את קובצי הקוד שהוספנו. תרגיל שלא יכלול את כל הקבצים הנדרשים, לא יתקבל וידרוש הגשה חוזרת (עם כללי האיחור הרגילים).

המלצה כללית: אחרי הכנת הקובץ להגשה, נעתיק אותו לתיקייה חדשה, נחלץ את הקבצים שבתוכו ונבדוק אם ניתן לפתוח את התיקייה הזו ולקמפל את הקוד. הרבה טעויות של שכחת קבצים יכולות להימנע על ידי בדיקה כזו.

**בהצלחה!**