

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

פרויקט סוף סמסטר

תאריכי הגשה: ראו להלן.

מטרות הפרויקט:

שילוב הידע שרכשתם במהלך השנה, בשני חלקי הקורס, לכדי פיתוח תוכנה מורכבת והתמודדות עם כל שלבי פתרון הבעיה, מניתוח הבעיה והגדרתה דרך תיכון הפתרון ועד השלמת המימוש.

תיאור כללי:

בפרויקט תממשו משחק כלשהו לפי בחירתכם. אתם יכולים לבחור כל משחק שעולה בדעתכם (**מלבד** הרשימה המופיעה להלן), אולם הוא צריך לענות על שני הקריטריונים הבאים:

1. רמת המשחק – על המשחק להיות מעניין ומהנה עבור המשחקים בו.
2. רמת הפתרון – על הפתרון להיות בעל תיכון ומימוש לא טריוויאליים.

דוגמאות לתיכון ומימוש לא טריוויאליים: מגוון דמויות ואינטראקציה ביניהם, לוח משחק מתוחכם, משחקים המשלבים גיאומטריה או פיסיקה מעניינת, שימוש באסטרטגיות (בינה מלאכותית), משחקים הדורשים ניהול יעיל במיוחד של משאבים (זמן ריצה או זיכרון).

רמת הקושי של הפרויקט תשוקלל בהערכת הפרויקט.

ניתן משקל רב יותר לאיכות הפרויקט מאשר לגודל שלו, כדי לא להפוך את הפרויקט לזולל זמן גדול מידי.

משחקים **שלא** ניתן לבחור:

Super Mario (כל הסדרה), Tower Defense, Sonic, Dave, Xonix, Tanks, Fireboy and Watergirl, Aladdin, Lion King, Gold Miner, Chicken Invaders, Bomberman, Packman, Digger, Lude runner, Pipe.

(רעיונות למשחקים ניתן למצוא למשל באתר: <https://www.crazygames.com>)

מימוש:

עליכם לשקול אלו מהטכניקות והשיטות שנלמדו במהלך השנה צריכות לבוא לידי שימוש בפתרון שלכם, בין השאר: תבניות (Templates), ירושה ופולימורפיזם, שימוש במבני נתונים ובאלגוריתמים של STL, שימוש ביכולות אחרות של הספרייה הסטנדרטית, שימוש בחריגות (Exceptions), שימוש בתבניות עיצוב (Design Patterns), שימוש בספריות חיצוניות כגון Boost וכו'.

גם ברמת התיכון חשוב להביא בחשבון כללים שנלמדו, ובפרט שווה להשקיע מחשבה בנושא של הפרדת התיכון והמימוש לחלקים שונים: תשתיות (כמו מבני נתונים הנדרשים למימוש), לוגיקה ולבסוף הצגה גרפית. הממשק הגרפי של המשחק ימומש בעזרת SFML, מן הסתם, אבל ניתן לבחור בדרכים אחרות כל עוד הקוד נשאר מונחה עצמים כראוי.

בשימוש בספריות חיצוניות, וודאו שניתן לקמפל ולהריץ את הפרויקט בלי צורך בהתקנות או הורדות נוספות. דרישה נוספת היא לשלב בתוכנה עזרה (help) עם הסבר לגבי אופן פעולת התוכנה, בפרט לגבי אפשרויות השליטה (למשל, הפעולות של מקשי המקלדת השונים או לחצני העכבר) ומטרת המשחק.

נקודות נוספות שחשוב להביא בחשבון:

- **אסטרטגיה** – אם המשחק שלכם הוא מול המחשב, כלומר אם יש דמויות או פעולות שנשלטות על ידי המחשב והוא "מחליט" מה לעשות איתן (למשל, השדונים בפקמן), רצוי שלמחשב תהיה לפחות אסטרטגיה בסיסית ולא שהכול יהיה אקראי.
- **פולימורפיזם ותבניות** – שקלו האם ראוי להשתמש בתבניות או בפולימורפיזם, ואולי בשניהם.
- **גנריות ואפשרות להרחבה** – שקלו את האפשרויות השונות למימוש גנרי של חלקים רלוונטיים. דוגמאות מצויות הן:

- קריאת לוח המשחק (אם יש כזה) מקובץ טקסט כדי לאפשר בקלות יצירת שלבים חדשים או שינוי של השלבים הקיימים.
- תיכון נכון של המחלקות כך שהוספה של יכולות חדשות, דמויות חדשות או חפצים חדשים תהיה פשוטה יחסית ותשתלב בקלות בקוד הקיים.
- שילוב נכון של מולטימדיה בתכנית כך ששינוי של תמונות, סאונד, אנימציה ואף החלפת ספרייה העוסקת בכך יהיו "נקיים" יחסית.

הערה טכנית:

נזכיר כי למרות ש-SFML מאפשרת הצגה של החלון להציג את החלון על מסך מלא (מצב fullscreen), המצב הזה בעייתי מאוד מבחינת הצגת הפרויקט, בפרט כשהיא מבוצעת דרך זום. אם בחרתם לתמוך במצב הזה בפרויקט שלכם, וודאו שניתן לשנות זאת בקלות (בלי שינויים בקוד!) כך שאפשר יהיה להריץ את המשחק גם בחלון רגיל.

הנחיות נוספות:

1. גדלי הצוותים יתואמו עם מורי הקורס, ההיקף והמורכבות צריכים להיות פרופורציונליים לגודל הצוות.
2. בדרך כלל, כל הצוות יקבל את אותו הציון לפרויקט חוץ מאשר מקרים חריגים בהם יהיה פער משמעותי בין חברי הצוות בהיקף התרומה לפרויקט. על כן חשוב שכל חברי הצוות ייטלו חלק פעיל בהצגה, בהבנת הפרטים ובהסברים.
3. הצגת הפרויקט היא כלי משמעותי עבורכם כדי להסביר ולהגן על עבודתכם, מהווה חלק עיקרי מהציון, וההשתתפות בה היא חובה.
4. לא ניתן לקבל דחיה להצגה או להגשת הפרויקט וימי האיחור והמתנה שהיו נהוגים בתרגילים אינם תקפים לפרויקט. למקרים חריגים באופן מיוחד יש לפנות למרצה לקבלת אישור (מראש!).
5. הערכת הפרויקט הינה הערכה ולא שקלול של נקודות. ההערכה מבוססת על שלושה אנשי צוות: מרצה, מתרגל, בודק תרגילים ומתייחסת למדדים הבאים: רמת המשחקיות, היקף ועומק, רמת התכנות מונחה העצמים, רמת התכנות הכללית, רמת ההבנה והבקיאות בפרטים.
6. לא ניתן לערער על ציון הפרויקט למעט מקרים חריגים של טעות בהערות או בשקלול. בדיקה חוזרת עשויה להעלות או להפחית את הציון.
7. ציון עבור פרויקט שעונה על מרבית הדרישות הוא 85. תוספות מהותיות ורמת תכנות גבוהה באופן משמעותי יוסיפו לציון. חוסר שלמות ובעיות אחרות יפחיתו מהציון.
8. הקוד של הפרויקט צריך להיות מקורי מתחילתו ועד סופו. שימוש בקוד כלשהו דורש אישור מראש.

לוחות זמנים:

כל ההגשות הן עד 23:59 בתאריך המצוין, כרגיל, אלא אם נאמר אחרת.

קמפוס הנביאים:

- בחירת המשחק וכלליו יום ג', כ"ז אייר, 4/6 (שלושה ימי עבודה" אחרי הגשת תרגיל 3)
- הגשת התיכון יום ה', כ"ט אייר, 6/6 (יומיים אחרי בחירת הפרויקט)
- הצגת השלד ימים ג'-ד', י"ט-כ' סיוון, 25-26/6 (שבועיים וחצי מהגשת התיכון)

- הגשת הפרויקט יום א', א' תמוז, 7/7 (שבוע וחצי מהצגת השלד)
- הצגת הפרויקט ימים ב'-ג', ב'-ג' תמוז, 8-9/7 (אחרי ההגשה!) – לא סופי, יעודכן בהמשך

קמפוס שטראוס-גברים:

- בחירת המשחק וכלליו יום ג', י' סיוון, 30/5 (שני "ימי עבודה")
- הגשת התיכון יום א', ט"ו סיוון, 4/6 (שלושה "ימי עבודה" אחרי בחירת הפרויקט)
- הצגת השלד יום א', כ"ט סיוון, 18/6 (שבועיים מהגשת התיכון)
- הצגת הפרויקט יום א', י"ג תמוז, 2/7 (כשבועיים מהצגת השלד) – ייתכנו שינויים
- הגשת הפרויקט יש להגיש עד ל-12 בצהריים ביום ההצגה
- הגשת תיקונים יום ג', ט"ו תמוז, 4/7 (יומיים לאחר הצגת הפרויקט)

ניקוד:

הניקוד מחולק עבור השלבים השונים כדלהלן:

- בחירת המשחק וכלליו 5%
- הגשת התיכון 5%
- הצגת השלד 10%
- הצגת הפרויקט והגשתו 80%

תיאור כללי של שלבי ההגשה:

בחירת הפרויקט והגשת חוקי המשחק – יש להעלות למשימה המתאימה ב-Moodle קובץ עם בקשה לאישור המשחק שבחרתם. בהודעה תכתבו את שם המשחק, תיאור קצר, ואם אתם מממשים משחק קיים שניתן לתת קישור להדגמה שלו באינטרנט, כדאי להוסיף קישור כזה. בנוסף, הוסיפו את תיאור כללי המשחק שאתם מתכוונים לממש והמגבלות הרלוונטיות.

הגשת תיכון – יש להעלות למשימה המתאימה ב-Moodle קובץ עם התיכון לבדיקה.

הצגת השלד – יש להציג את שלד הפרויקט, כלומר הקוד שנכתב עד כה, בפני המתרגל/ת במהלך זמן התרגול והסדנה. (על השעות המדויקות תבוא הודעה בהמשך).

הצגת הפרויקט – הצגת הפרויקט לסגל הקורס (בדומה למה שהיה בסמסטר א'). שימו לב, בזמן הצגת הפרויקט אתם קרובים מאוד לשעת ההגשה בפועל, ולכן הפרויקט צריך להיות מוכן או כמעט מוכן כך שהשינויים אחרי ההצגה יהיו מינוריים לכל היותר (בקמפוס הנביאים, ההצגה היא אחרי הגשת הפרויקט, כך שהפרויקט כבר

בצורתו הסופית בכל מקרה). הודעה סופית על מועד הצגת הפרויקט תבוא בהמשך. **ההשתתפות בהצגת הפרויקט היא חובה!**

הגשת הפרויקט – הגשת הפרויקט כרגיל על פי ההנחיות הרגילות. כמו כן, העלו ל-Youtube סרטון הדגמה, והעלו למשימה המתאימה (בנוסף לפרויקט עצמו) תמונה של צילום מסך מייצג וקישור לסרטון.

פירוט שלבי ההגשה:

בחירת הפרויקט והגשת חוקי המשחק:

את הפרטים המוסברים להלן, יש לכתוב בקובץ docx (קובץ Word) או קובץ PDF. יש להגיש קובץ יחיד, בשם oop2_project_rules-firstname_lastname (עם הסיומת המתאימה לפי סוג הקובץ), כאשר firstname_lastname הוא השם המלא (לדוגמא אלברט איינשטיין, אם יגיש קובץ PDF יגיש כך: oop2_project_rules-albert_einstein.pdf). במקרה של הגשה בזוג, שם הקובץ יהיה לפי התבנית oop2_project_rules-firstname1_lastname1-firstname2_lastname2, עם שמות המגישים בהתאמה (ללא רווחים; כלומר, בשמות עצמם יש להחליף רווחים בקו תחתי ובין שמות חברי הצוות השונים יפריד מקף, כפי המודגם לעיל), וכן הלאה. במקרה של הגשה בצוות, **רק אחד** מהמגישים יגיש את הקובץ ולא כולם. אם בכל זאת יש צורך בהגשת יותר מקובץ אחד, יש לדחוס אותם לקובץ zip בשם הנ"ל. במקרה כזה, וודאו כי קובץ ה-zip מכיל תיקייה ראשית אחת, ורק בתוכה יהיו כל הקבצים המוגשים.

הקובץ צריך לכלול את הפרטים הבאים:

- שמות המגישים ומס' ת.ז.
 - שם המשחק
 - תיאור קצר של המשחק
 - קישור לדוגמה של המשחק (אם יש כזו)
 - הכללים שאתם מתכוונים לממש ומגבלות שתשימו לצורך פישוט המימוש
- אם מדובר במשחק לא מוכר ואין לינק לדוגמה, חשוב להשקיע ולפרט יותר בתיאור המשחק.

דוגמה להודעה על בחירת המשחק וכלליו:

מגישים:

ישראל ישראלי 0123456789

יצחק יצחקי 0987654321

שם המשחק: פקמן

תיאור:

לוח המשחק הוא מבוך שבו פקמן מסתובב, אוסף נקודות ומתחמק מהשדונים שמנסים לתפוס אותו. על השחקן לאסוף את כל הנקודות כדי לסיים בהצלחה את השלב. לפקמן מספר מוגבל של פסילות עד

שהמשחק מסתיים. בכל פעם שרוח רפאים תופסת את פקמן, יורדת לו פסילה אחת. לסוגים שונים של נקודות יש השפעות שונות על פקמן, חלק משפיעות על המהירות שלו (ממהרות או מאטות אותו), וחלק מאפשרות לו לאכול את רוחות הרפאים.

קישור: [פקמן](#)

כללים ומגבלות:

במשחק יהיו שני סוגים של שדונים.

במשחק תהיה אפשרות לשחקן אחד בלבד.

למשחק יהיו שלושה שלבים. ההבדל בין השלבים מתבטא בצורת המבוך, ובהגדלת מספר השדונים החכמים לעומת השדונים האקראיים.

לא ניתן יהיה לשמור את המשחק ולהמשיך אותו בפעם אחרת, אולם ניתן יהיה לעצור את פעולת המשחק לזמן מה (pause) ולהמשיך אח"כ כל עוד לא סגרנו את המשחק.

תהיה טבלת שיאים (high score).

השדונים החכמים יממשו אלגוריתמים מתחום תורת הגרפים כדי לעקוב אחרי פקמן בצורה יעילה.

השדונים האקראיים, ינועו באופן אקראי.

בתחום העוגיות המיוחדות נממש רק את אלה המשפיעות על מהירות השחקן, אך לא את אלה

המאפשרות לו לאכול את השדונים.

הגשת תיכון:

כמו בסמסטר א', הגשת התיכון נעשית בהגשת דיאגרמת UML המציגה את המחלקות המתוכננות עם מאפיינים ופונקציות עיקריות שלהן והקשרים בין המחלקות (ירושה, הכללה), ובנוסף הסבר מילולי קצר שיסייע בהבנת הדיאגרמה וחלוקת האחריות בין המחלקות. אם מחלקה היא תבניתית, חשוב לציין זאת.

גם כאן, כפי שאמרנו בעבר, על התיכון להיות סביר והגיוני בהתאמה למורכבות המשחק, אולם ברור מראש שיש סיכוי גדול שבמהלך המימוש תשנו את התיכון.

לגבי דרך הגשת התיכון, ראו בהוראות מסמסטר א' הצעות כיצד לייצר דיאגרמה כמבוקש. את ההסברים הנוספים יש לצרף בקובץ Word (בעברית).

את הקבצים להגשה כווצו לקובץ zip, כרגיל, כשהשם שלו כעת בנוי בתבנית הבאה: oop2_design-firstname_lastname.zip, עבור מגיש בודד, כאשר firstname_lastname הוא השם המלא (לדוגמא אלברט איינשטיין יגיש את התיכון כך: oop2_design-albert_einstein.zip). במקרה של הגשה בזוג, שם הקובץ יהיה לפי התבנית oop2_design-firstname1_lastname1-firstname2_lastname2.zip, עם שמות המגישים בהתאמה (ללא רווחים; כלומר, בשמות עצמם יש להחליף רווחים בקו תחתי ובין שמות חברי הצוות השונים יפריד מקף, כפי המודגם לעיל), וכן הלאה. במקרה של הגשה בצוות, **רק אחד** מהמגישים יגיש את הקובץ ולא כולם.

וודאו כי קובץ ה־zip מכיל תיקייה ראשית אחת, ורק בתוכה יהיו כל הקבצים המוגשים.

הצגת השלד:

בדומה לסמסטר א', נצפה לראות כבר שלד של המשחק עם התקדמות משמעותית ולא רק התחלה בסיסית של כתיבת הקוד.

יצירת תמונה וסרטון הדגמה:

הפרטים מוסברים בקובץ נפרד.

הגשת הפרויקט

בדומה להגשת התרגילים, וכפי המוסבר בסעיפים הבאים.

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. תיכון (design): הסבר קצר מהם האובייקטים השונים בתוכנית, מה התפקיד של כל אחד מהם וחלוקת האחריות ביניהם ואיך מתבצעת האינטראקציה בין האובייקטים השונים.
 5. רשימה של הקבצים שיצרנו, עם הסבר קצר (לרוב לא יותר משורה או שתיים) לגבי תפקיד הקובץ.
 6. מבני נתונים עיקריים ותפקידיהם.
 7. אלגוריתמים הראויים לציון.
 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 ומתחתיה 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-VS 2022".
אנא השתמשו בדרך זו (אחרי שהגדרתם כראוי את שמות הצוות ב-MY_AUTHORS: **נשים את שמות המגישים בתוך המרכאות, נקפיד 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_project-firstname_lastname.zip (או במקרה של הגשה בזוג – oop2_project-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 תחליטו 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!