**דף הכנה למתכונת והמבחן חיצוני**

חלק א'- דגשים למבחן

חלק ב'- נקודות חשובות על האלגוריתם המרכזי בפרויקט

חלק ג'- שאלות אפשריות למבחן

**חלק א'**

1. עליכם לדעת היטב כל דבר בפרויקט.
2. ביום המתכונת יש להגיש ספר פרויקט עדכני + פרויקט סופי. יש להביא את הספר (בפורמט וורד וגם בפורמט pdf)והפרויקט במייל ובדיסק און קי.
3. הפרויקט צריך להיות מתועד, עובד, עם תפריט (ריסטרט, הוראות, יציאה).
4. רצוי ליצור קובץ jar(קובץ הרצה) לפרויקט. ההסבר נמצא בסוף מסמך זה.
5. **חובה לדעת להסביר את האלגוריתם הראשי בפרויקט, את הסיבוכיות שלו ולציין את כל מבני הנתונים שהשתמשתם בהם בפרויקט.**
6. אתם צריכים לדעת להסביר מונחים שלמדנו במהלך השנה, כמו: הכלה, ירושה, פולימורפיזם, תהליכונים (Thread), גנריות.
7. מומלץ להתחיל להציג את הפרויקט תוך כדי שמשחקים בו, ואז להסביר את האלגוריתם הראשי בלי שישאלו. כמה שיותר תסבירו שוטף, ככה פחות ישאלו שאלות.

**חלק ב'**

**נקודות חשובות על דייקסטרה**שמומלץ להבין ולדעת (לפי האלגוריתם שלמדנו בכיתה). אתם צריכים לדעת להסביר את דייקסטרה בעל פה ושוטף!

* אלגוריתם למציאת מסלול קצר ביותר בין שני קודקודים בגרף. אצלנו זה כדי למצוא את המסלול הקצר ביותר בין האויב לשחקן האנושי...
* האלגוריתם עובר קודקודקודקוד לפי סדר מסוים (לפי המרחק המינימלי...), ולכל קודקוד הוא עובר על כל השכנים שלו. עבור השכן, הוא בודק איזה ערך קטן יותר: המרחק הנוכחי שלו מקודקוד המקור, או המרחק של הקודקוד ממנו הגענו + המשקל על הקשת בניהם. המרחק הקטן יותר נשמר, וגם הקודקוד ממנו הגענו.

דוגמא: עבור קודקודX עוברים על כל השכנים (נניח Y,Z), אז עבור שכן Z נבדוק איזה מרחק קטן יותר- מרחקו הנוכחי מקודקוד המקור, או המרחק של Xמקודקוד המקור + המשקל על הקשת בין X לZ.

* אלגוריתם דייקסטרה מכיל שתי מטודות: אחת שמקבלת קודקוד מקור, ולכל קודקוד בגרף היא מחשבת את המרחק הקצר ביותר ממנו לקודקוד המקור, והמטודה השנייה מקבלת קודקוד יעד ומחזירה את המסלול הקצר ביותר לקודקוד המקור (עבור קודקוד היעד נשמר מאיזה קודקוד הגענו אז הולכים אחורה ואחורה בקודקודים עד שמגיעים ליעד וזה המסלול שלנו).
* מומלץ לצייר גרף ולהריץ עליו את האלגוריתם כדי להבין אותו היטב.
* סיבוכיות|E|+|V|Log|V|, זו הסיבוכיות של דייקסטרה כשמשתמשים בתור קדימויות ( |E| מסמל כמות של קשתות, |V| מסמן כמות קודקודים. הוצאה מתור קדימויות היא לוג V).
* מבני נתונים שהשתמשנו בהם בשביל הדייקסטרה: גרף (שמורכב בעצם ממטריצת קודקודים שבניהם יש קשתות), ותור קדימויות.

**נקודות חשובות על מינימקס**שמומלץ להבין ולדעת (לפי האלגוריתם שלמדנו בכיתה). חייבים לדעת להסביר את מינימקס בעל פה ושוטף!

* אלגוריתם למציאת המהלך הטוב ביותר לשחקן המחשב במשחקי לוח.
* עבור תור מחשב מסוים, המינימקס מוצא את כל המהלכים האפשריים של המחשב, עליהם הוא פורש כתגובה את כל המהלכים האפשריים של השחקן האנושי, עליהם את כל המהלכים האפשריים של... וככה עד שמגיעים לעומק מינימקס הרצוי.  
  מחשבים את הציון ללוח שקיבלנו, ואז מעלים למעלה את הציון- כל פעם בוחרים ציון מינימום או מקסימום תלוי אם זה תור של שחקן מחשב או אנושי. בסופו של דבר המחשב בוחר את המהלך הטוב ביותר בשבילו.
* סיבוכיות , כאשר B הוא גורם ההסתעפות של העץ (כלומר, כמה מהלכים אפשר לפרוש ממהלך מסוים), וD הוא עומק המינימקס.
* מבנה נתונים שהשתמשנו בו היה רשימה מקושרת (ששימשה לצורך עץ בעזרת רקורסיה).

חלק ג'

**אלה שאלות שיכולים לשאול אתכם במבחן. יכול להיות שלא ישאלו אתכם כלום! הכל תלוי באיך תציגו את הפרויקט. כדאי להסביר בקצרה (מאד בקצרה) מה המשחק, מה האלגוריתם, מה הסיבוכיות ומבני נתונים, זה כבר מראה לבוחן מה אתם יודעים.**

הבוחן שואל מה שנראה לו באותו רגע אז תהיו מוכנים לזה ואם אין לכם תשובה תנסו להיזכר או לבקש הבהרה לשאלה.

שאלות אפשריות כלליות:

* הסבר על מושג.

1. **פולימורפיזם** -הוא היכולת להתייחס להתייחס לעצמים מטיפוסים שונים (צורות שונות) באופן אחיד. לדוגמא ENEMY ו PLAYER , לשניהם אני יכול לתת פקודה של תזוזה.
2. **ירושה** - ירושה היא מנגנון המאפשר להגדיר את המשותף שבין מספר מחלקות במחלקה אחת. יורשת את כל מאפייני המחלקה והמטודות של מחלקת האב.למשל PLAYER שיורש מ MOVING UNIT
3. **תהליכון** - מבצע פקודות במקביל לתהליכונים אחרים. כיוון שאין מקביליות אמיתית (אלא אם כן יש כמה מעבדים) המערכת הפעלה מכניסה בתור את התהליכונים לביצוע ככה שנראה שיש מקביליות . היתרון שיש פקודות הדורשות קלט מבן אדם ועד שמקבלים אותו התוכנית תקועה. למשל כמה טיימרים שרצים במקביל
4. **הכלה** - כאשר מחלקה אחת מכילה משתנה מחלקה מסוג מחלקה אחרת. מה שמאפשר לחלק תכונות בצורה נכונה , למשל מחלקת שחקן אשר דואגת לתזוזה ויריות והמחלקה UNIT דואגת למד החיים ולמיקום הלוגי של השחקן.
5. **גנריות** - תכנות גנרי מאפשר לנו להגדיר מטודות ומחלקות המטפלות באובייקטים מטיפוסים לא ידועים. למשל LINKED LIST
6. **מחלקה\מטודה מופשטת** - מחלקות מופשטות (abstract) הן מחלקות שאי אפשר ליצור מהן אובייקטים באופן ישיר.הן מגדירות אוסף של תכונות המשותפות לכמה מחלקות, ומשמשות רק כבסיס ליצירת מחלקות אחרות.
7. **ממשק** - (interface)הוא כלי המאפשר להוסיף פונקציונליות משותפת למחלקות שונות שאינן נמצאות בהיררכית ירושה משותפת.

* מה ההבדל בין תהליכון לתהליך

תהליך זה כל הPROCESS של המשחק או כל EXECUTABLE שרץ במקביל לכל שאר התהליכים ומנוהל על ידי מערכת ההפעלה. תהליכון זה THREADS שנוצרים ויש להם גישה המוגבלת ל PROCESS שאחראי עליהם ומנוהלים בעזרת מערכת ההפעלה וגם על ידי המתכנת בעזרת מוניטורים וJOIN . ההבדל הוא בניהול ובסביבת הכרה.

* מה זה protected ומה ההבדל בינו לבין private

protected מאפשר גישה לבנים שיורשים מהמחלקה הזאתי כאשר ב private אין גישה ישירה אלא אם כן יש GET\SET. ב ג'אווה הprotected גם נותן גישה לכל הPACKAGE.

* איזה מטודות ניתן לדרוס (virtual)

כל מטודה פרט למטודות FINAL ו STATIC וגם מטודות PRIVATE כיוון שאין גישה אליהם מהירישה.

* איך הפרויקט מפריד בין לוגיקה לגרפיקה

לכל דבר מצויר יש לי מחלקה נוספת המכסה את הלוגיקה . לדוגמא PROJECTILEG ו PROJECTILE.

* איזה תהליכונים יש בפרויקט (טיימרים...)

יש לי טיימרים על הדמויות שזזות , על הכדורים ועל הזמן עד לתקיפה. חלקם פשוטים מאוד שפשוט סופרים עד כל כמה זמן ואז מפסיקים כשמגיעים ל0 והאחרים בודקים אופציות תזוזה ולוקחים תזוזה כזאתי בהתאם לפי הטיימר.

* איזה אירועים יש בפרויקט (מוכנים כמו מקלדת ועכבר וכאלה שהוכנו לבד)

יש לי אירועים של מקלדת למשל כפתור ESC והמקשיב זה המפה. יש מחלקה השומרת את כל המפתחות על המקלדת וכאשר כפתור נלחץ היא מעדכנת אצלה את הלחיצה וכל מי שמשתמש בקלט מהמשתמש פונה למחלקה הזאתי ובודק האם כפתור מסוים דלוק.יש לי כפתורים שמודיעים לפריימים מסוימים להופיע וללכת אשר מוכנים כבר.

* איזה ממשקים יש בפרויקט

ממשק התקפות אויב אשר מודיעה להתקפה עצמה מתי האנימציה נגמרה.  
 ממשק לחיצת עכבר בעיצוב המודיע לקלאס של הגרפיקה של השחקן.

ממשק של דמויות שיכולות לנצח,מכיל בתוכם פונקציות של נצחון ובדיקות.

ממשק של ציור , כל גרפיקה שתופיע על המסך תממש את הממשק הזה

ממשק של כפתור ESC לצורך סגירת חלון המשחק ופתיחת הMENU.

ממשק של גרף קודקוד וקשת , לצורך אחידות בין כל הפרויקטים.

* מה היה הכי מאתגר בפרויקט

הכי מאתגר היה לכתוב את הקישור בין קירות , אם תשימו לב הקירות מחוברות וכשנהרס ליד הקירות משתנות , זה כי יש 16 סוגי כיוון לכל קיר.לשאול 16 פעם על כל קיר זה גוזז הרבה מהמעבד ולכן חיפשתי משהו פשוט יותר, כיוון שבתמונות עצמן אין סדר לוגי כל שהו הייתי צריך להמציא אותו לדעת לשאול תנאים מתאימים שיכסו את כל האופציות בכמה שפחות שאלות.

* מה למדת בזכות הפרויקט (בג'אווה)

למדתי איך לעבוד עם תמונות כמו למשל הפיכה , מראה ,הזזה בזווית וגזירה של תמונות.למדתי עד כמה חשובים כמה משיטות המפעל כמו סינגלטון ואיטרטור.

* מה מעניין/מיוחד בפרויקט

הייתי אומר שהאווירה במשחק אף פעם לא תהיה אותו דבר כמו מקודם כי המפה נוצרת ראנדומלית כל פעם ונותנת ציורים שונים גם. זה נעשה ב RANDOMMAPCREATOR

* איזה ירושה יש בפרויקט

יש ירושה של שחקן היורש מגוף זז (אבסטרקטי) שיורש מאובייקט על מפה שיורש מפאנל. כל חלון יורש מ JFRAME וכל פאנל יורש מ JPANEL

* איך התפריט עובד

בתפריט אפשר להתחיל משחק ,לבדוק איך לשחק ,לעצב את הדמות או לצאת מהמשחק. בסוף כל משחק ניתנת אופציה לעשות ריסאטארט או ליצור משחק חדש (אם ניצחת)

שאלות אפשריות לפרויקט מפות:

* מהו קובץ XML

קובץ XML בנוי כדי לשמור ולהעביר מידע , קובץ XML בנוי כדי להיות קריא על ידי בני אדם וגם על ידי מכונה.

* איך קובץ הXML בנוי

התגית החיצונית היא MAP אשר מחזיקה את כל המטריצה. אחריה יש תגיות של LINE אשר קובעות שורה ואחריה יש תגיות של AREA אשר קובעות עמודה. בכל AREA יש ערך אשר הוא הערך של הIMGID שחשבתי עליו מראש.

* מהו הייצוג הגרפי של המפה

במפה יש את המסגרת שאי אפשר לעבור והיא עשויה מקירות שלא ניתן להרוס .

* מה קורה אחרי שטוענים את קובץ הXML למשחק

אחרי שאני טוען את שני קבצי XML למשחק ( קיר ורצפה) יש במטריצה של MAPG את כל הבלוקים מוגדרים ומוקצים ומוכנים לשימוש .ישר אחרי זה מתחיל הציור על ידי הPAINT של הפאנל אשר מצייר את המפה בהתאם.

* איך השחקן זז

השחקן זז בעזרת לחיצת מקשים אשר מפעילים פונקציות תזוזה בזמן שהטיימר של השחקן מתקתק. הפונקציות האלא מזיזים את השחקן במפת פיקסלים וגם במפה הלוגית.

* איך האויב זז (דייקסטרה...)

האויב זז באותה שיטה כמו השחקן , אך הוא לא מקשיב לקלט מהשחקן אלא לאלגורטים דייקסטרה שמחזיר לו מספר האומר איזה כיוון לבחור (שהוא הקצר ביותר עד השחקן)

* איך בודקים התנגשויות עם בלוקים/קווים במפה

יש לי שני בדיקות , אחת מהם היא בעזרת יצירת מלבנים בגודל ה PANEL ולבדוק המלבנים מתנגשים בעזרת INTERSECTS. השנייה בודקת האם המרכז של אובייקט מסוים נוגע באובייקט אחר , ככה מאפשר לאובייקטים להגיע עד האמצע ואז לפגוע.

* איך מתבצעות היריות

על ידי לחיצת מקשי כיוון בטיימר של השחקן יש פונקציות אשר מקצות יריות עם כיוון וזווית של התמונה עצמה (כדי להפוך אותה לכיוון שיורים אותה) היריה עצמה היא טיימר אשר מקדמת את עצמה ובודקת אם יש התנגשות מבחינת אמצע עם הקירות או אויב.

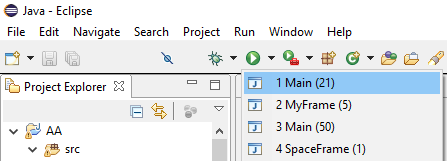
שאלות אפשריות לפרויקט לוח:

* איך עשית רמות בפרויקט

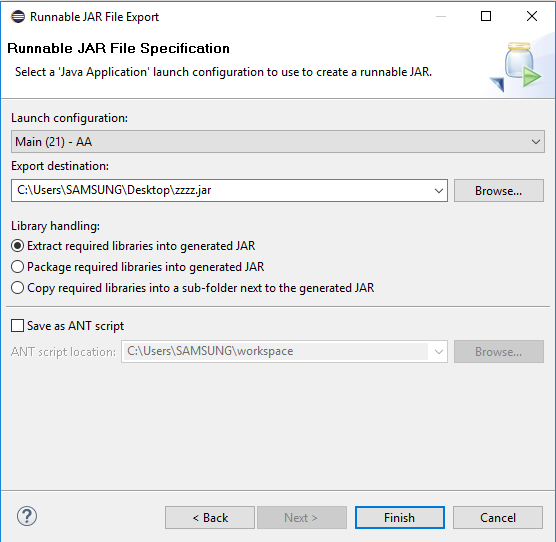
המשחק נהיה קשה יותר ככל שמנצחים יותר את האויב וקל יותר ככל שיותר מפסידים לאויב. את כמות ההפסדים \נצחונות אפשר לראות במסך המשחק למעלה,כאשר האדום זה השחקן וכחול זה האויב.

**הנחיות ליצירת קובץ הרצה למשחק:**

1. **מריצים את המשחק וסוגרים**
2. **ליד הכפתור הרצה יש משולש, לוחצים עליו ובודקים מה שם הקובץ שרץ אחרון**

****

1. **ניגשים ל**File->Export->Java->Runnable JAR File
2. **בוחרים את הקובץ שרץ אחרון, מסמנים את תיבות הבחירה לפי התמונה ולוחצים finish**



1. **ייתכן ותקפוץ הודעת שגיאה כלשהי. זה לא נורא.**
2. **מנסים להריץ את קובץ הjar שנוצר במקום שבחרתם. אם זה לא עובד, צריך להדביק ליד הקובץ את התיקיות Images וFiles אם יש.**
3. **מריצים שוב. עכשיו זה אמור לעבוד.**

לפעמים זה לא עובד וצריך לטרוח קצת יותר ולשנות את מיקום התמונות. לא חייב שיהיה קובץ JAR, אפשר להגיש בלי, אבל מי שרוצה בכל זאת שישלח לי מייל ואני אתן לו הוראות איך לעשות את זה.

למי שעובד לו הJAR אז **שיעדכן בספר פרויקט שהוראות התקנה והפעלה** זה רק להעתיק את התיקייה שמכילה את הקובץ הזה למחשב, ללחוץ עליו דאבל קליק והמשחק יפתח.