**README – ex04**

**שם הקורס**: תכנות מונחה עצמים ופיתוח משחקים.

**תרגיל:ex04**

**שמות הסטודנטים**:

* שירה עטווד | ת"ז 311601199
* מיכל בוקס | ת"ז 205980725
* אליעד קרני | ת"ז 206329104
* יאיר דור | ת"ז 318169505

**הסבר כללי:**בתרגיל זה, ממשנו משחק רשת לוקאלי של מירוץ מכשולים.

כללי המשחק הם:

על השחקנים לצאת מצידה השמאלי של המפה ולגעת עם השחקן בנקודת הסיום שנמצאת בצידה הימני.

תנועת השחקנים נעשית ע"י לחצני החצים או מקשי WSDA, מקש רווח לקפיצה ולחיצת העכבר ליריית טיל (אם השחקן אסף מתנה כפי שיוסבר בהמשך). לאורך המפה מפוזרות מתנות אשר ב-3 נגיעות בהן נעלמות, האדם אשר נגע במתנה בפעם השלישית מקבל בונוס של יכולת שיגור טיל בודד תוך הקשת העכבר בכיוון הטיל הרצוי (השחקן יקיש על הנקודה אליה ירצה לשגר את הטיל). בכדי לטפס על חבל תחילה על השחקן להיתלות עליו וללחוץ את כיוון הטיפוס הרצוי (למעלה או למטה), בכדי לעזוב את החבל על השחקן לקפוץ ממנו לכיוון הרצוי. היה ונפל השחקן לתחתית המפה או דרך על מכשול רצפה בזמן שהופעל, ישוגר השחקן לנק' הציון האחרונה בה נגע, אם לא נגע בנקודת ציון כלל, ישוגר השחקן לנק' התחלת המירוץ. לאחר השיגור, ישותק השחקן למשך 3 שניות. השחקן הראשון שיגע בנק' הסיום ינצח במירוץ.

במפה מפוזרים:

* בלוקים עליהם יכולו השחקנים ללכת.
* בלוקים נעים אשר שימושם דומה.
* נק' ציון אשר במגע השחקן בה, אם יפול השחקן ישוגר אליה.
* מכשולי רצפה אשר בכל זמן קצוב מופעלים, ואילו נגע בהם השחקן ישוגר כמצוין לעיל.
* קשתים אשר יורים חץ (או טיל או כדור אש בהתאם לסוג המפה שנבחר) אילו יגע השחקן בחץ, ידחף.
* מתנות כמצוין לעיל.
* חבלים איתם יכול השחקן לטפס למעלה ולמטה במפה.
* אבנים נופלות: מכשול אשר כל זמן קצוב נופלות מתקרת המפה.
* נק' סיום אשר השחקן הראשון שיגע בה, ינצח במירוץ.

ישנה אפשרות משחק של שחקן יחיד בה השחקן ישחק לבד ללא התחברות לרשת.

**תיכון :**

הוגדרו מחלקות:

* אנימציה:
* קשת: מכשול משחק היורה אובייקט משחק פרוזקטייל כל זמן קצוב למטרה קבועה במפה.
* כפתור בסיסי: כפתור ויזואלי המקבל את מיקומו וגודלו.
* בלוק: רצפה קבועה
* לוח משחק:
* כפתור: לחצן ויזואלי היורש ממחלקת כפתור בסיסי הנועדה להעביר כנדרש את המשתמש למצב הבא אליו הוא נועד להוביל.
* נק' ציון: מחלקה יורשת מאובייקט משחק דינאמי. בהתנגשות השחקן נרשמת נק' הציון אצל השחקן ובשיגור השחקן ישוגר לנק' זו.
* קליינט: אובייקט היורש ממחלקת אובייקט תקשורתי, המחלקה מנהלת את התקשורת של המשחק עם הסרבר (מארח).
* מנהל התנגשויות: מחלקה המנהלת (ע"י שיטת "מולטי מתוד") את התנגשויות האובייקטים במשחק תוך שיתוף פעולה עם אובייקט מסוג חיישן התנגשויות.
* חיישן התנגשויות: מתריע על התנגשויות אובייקטים בעולם הפיזיקאלי.
* קונטרולר: מנהל את אובייקטי המשחק כנדרש.
* פאקטורי: מחלקה המייצרת אובייקטים כפי שנלמד בשיטת "פאקטורי" בהרצאות
* אבן נופלת: מחלקה יורשת מאובייקט דינאמי. בכל זמן קצוב, אובייקט המחלקה יפול מנק' ההתחלה ועד להתנגשות האובייקט עם אובייקט סטטי,

אם האובייקט אינו פעיל לזמן מסויים. הוא יחזור חלילה על פעולתו.

* מכשול רצפה: אובייקט יורש ממחלקת אובייקט דינאמי.
* אובייקט משחק: מחלקה אבסטרקטית של אובייקטי המשחק לסוגיהם.
* מתנה: מחלקה יורשת מאובייקט דינאמי. בפגיעת שחקן 3 פעמים באובייקט השחקן יקבל בונוס כלעיל.
* אבן זזה -
* אובייקט תקשורתי- מחלקה אבסטרקטית הדואגת לשליחה וקבלת הודעות ושמירת נתוני השחקן.
* מפת זוגות – טבלת גיבוב המקבלת מפתח של זוג משתנים. (הספרייה הסטנדרטית אינה מספקת טבלת גיבוב כזו אלה רק טבלת גיבוב של משתנים פרמיטיביים)
* שחקן – מחלקת שחקן המשחק.
* Projectile.h/cpp -
* מחלקת מקורות -
* מחלקת חבל -
* מחלקת שרת -
* מחלקת אובייקט סטטי-
* מחלקת מכשול -
* מחלקת בחירת סוג משחק-
* מחלקת משחק במצב משתתף-
* מחלקת מצב בחירת משחק-
* GameState.h/cpp -
* HowToPlayState.h/cpp -
* LobbyState.h/cpp -
* MainMenu.h/cpp -
* MenuState.h/cpp -
* MultiplayerMenuState.h/cpp -
* NetworkGameState.h/cpp -
* ServerGameState.h/cpp -
* StandardMenuState.h/cpp -
* State.h/cpp -
* StateManager.h/cpp

**רשימת קבצים :**

* main.cpp - הפונקציה המרכזית. מייצרת אובייקט מסוג Controller ומריצה את המשחק.
* Animation.h/cpp – קובץ מחלקת אנימציה.
* Archer.h/cpp – קובץ מחלקת קשת.
* BaseButton.h/cpp – קובץ מחלקת כפתור בסיס.
* Block.h/cpp – קובץ מחלקת בלוק
* Board.h/cpp – קובץ מחלקת לוח משחק.
* MovingBlock.h/cpp – קובץ מחלקת אובייקט דינאמי
* NetworkObj.h/cpp – קובץ מחלקת מנהל תקשורת.
* PairHash.h – קובץ מחלקת טבלת גיבוב זוגות.
* Player.h/cpp – קובצי מחלקת שחקן.
* Projectile.h/cpp -
* Recources.h/cpp -
* Rope.h/cpp -
* Server.h/cpp -
* StaticObj.h/cpp -
* StaticObstacle.h/cpp -
* ChooseBoardState.h/cpp -
* ClientGameState.h/cpp -
* GameMenuState.h/cpp -
* GameState.h/cpp -
* HowToPlayState.h/cpp -
* LobbyState.h/cpp -
* MainMenu.h/cpp -
* MenuState.h/cpp -
* MultiplayerMenuState.h/cpp -
* NetworkGameState.h/cpp -
* ServerGameState.h/cpp -
* StandardMenuState.h/cpp -
* State.h/cpp -
* StateManager.h/cpp
* Button.h/cpp -
* CheckPoint.h/cpp -
* Client.h/cpp -
* CollisionHandler.h/cpp -
* ContactListner.h/cpp -
* Controller.h/cpp -
* Factory.h/cpp -
* FallingBlock.h/cpp -
* FloorObstacle.h/cpp -
* GameObj.h/cpp -
* Gift.h/cpp -
* Macros.h -
* MessagesStruct.h/cpp -
* MovingBlock.h/cpp -
* NetworkObj.h/cpp -
* PairHash.h -
* Player.h/cpp -
* Projectile.h/cpp -
* Recources.h/cpp -
* Rope.h/cpp -
* Server.h/cpp -
* StaticObj.h/cpp -
* StaticObstacle.h/cpp -
* ChooseBoardState.h/cpp -
* ClientGameState.h/cpp -
* GameMenuState.h/cpp -
* GameState.h/cpp -
* HowToPlayState.h/cpp -
* LobbyState.h/cpp -
* MainMenu.h/cpp -
* MenuState.h/cpp -
* MultiplayerMenuState.h/cpp -
* NetworkGameState.h/cpp -
* ServerGameState.h/cpp -
* StandardMenuState.h/cpp -
* State.h/cpp -
* StateManager.h/cpp –
* הודעת אודות משתתף
* הודעת משתתף
* הודעת הוספת משתתף
* הודעת עדכון אובייקט דינאמי
* הודעת עדכון סנכרון
* הודעת יריית שחקן
* הודעת תחילת משחק

**מבני נתונים עיקריים:**

* מנהל מצבים אשר מעביר את המשחק בין המצבים כרצוי (מצב תפריט, מצב קליינט, מצב סרבר, מצב משחק וכו')
* Unordered\_maps סטטיות המוכלות בסינגלטון ומחזיקות את הנתונים שנפתחו מקבצים חיצוניים.
* עולם פיזיקאלי המחשב את הדמיית חוקי הפיזיקה של המשחק.
* חיישן העולם הפיזיקאלי המזהה ומתריע על התנגשויות בעולם הפיזיקאלי
* מנהל התנגשויות אשר בשיתוף בעולה עם חיישן העולם הפיזיקאלי מנהל את סוגי ההתנגשויות לסוגיהן.
* לוח משחק המנהל את הפעלת האובייקטים השונים במשחק כנדרש והצגתם במסך.
* מנהל תקשורת אשר דואג לקבל או לשלוח הודעות על אודות המשחק בכדי לסנכרן בין השחקנים המשתתפים בו.
* וקטור של אובייקטים זזים המתעדכנים ע"י השרת.
* וקטור של אובייקטים סטטים.

**אלגוריתמים הראויים לציון:**

* חישובים פיזיקאלים תוך נירמול וחישובי זוויות וקטורים.
* ניהול התנגשויות ע"י שיטת מולטי מטוד.
* קריאה וייצור אובייקטים ע"י שיטת פאקטורי.
* קריאה ושליחת הודעות ע"י פונקציות טמפלטיות. בכל הודעה יש בהתחלה את תיאור סוג ההודעה ולאחר מיכן את תכולת ההודעה.
* סינכרון כל האובייקטים הדינאמיים במשחק פרט לסוג אחד.
* ניהול מצבים ע"י שילוב שיטת commad ושיטת state manager.

**באגים ידועים:**

* המשחק בנוי לרזולוציית מסך 1920 x 1080.
* מידי פעם, הקליינטים לא מצליחים למצוא את המארח, במקרה זה על המארח לחזור לתפריט הראשי ולארח משחק מחדש.

**הערות אחרות:**

* מכשולי הרצפה אינם מסונכרנים בין השחקנים.