A screenshot of a computer

Description automatically generated

niso,inbarbahnof

207447970,207009721

1) להעלמת לבנים בחרנו בשיטה 2, החלטנו שהעלמת לבנה היא תכונה ייחודית למנהל המשחק, כלומר אנחנו לא רוצים לתת לשאר המחלקות גישה לכל האובייקטים האחרים במשחק. בכך, היינו מוכרחים להעביר את המשחק לשאר המחלקות ולתת להם גישה רק לשיטה שמורידה אובייקט ספציפי – לה הם יכולים לשלוח רק את עצמם. שיטה זו מחזקת את עיקרון האנקפסולציה. כמובן שיש בכך חסרונות שמחלקות אחרות בעלות גישה לזה אך שיטה זו הייתה המועדפת על ידינו.

2) החלטנו שהוספה והורדה של חיים זה חלק קריטי מהתנהלות המשחק, ולכן המימוש של יצירת הקאונטר, הורדת חיים והוספת חיים קורה בBrickGameManager - גם המימוש וגם הקריאה. מבחינת סדר, פיצלנו את הפונקציונאליות למס' פונקציות כדלקמן:

• MakeHeartLife – which is in charge of graphic life represnstaion -

• - MakeLifeCount – which is in charge of graphic life counter representation

• - UpdateLifeGraphic – updates the coloring depending on the amount of lifes

• - כמו כן, כמות החיים שמורה במשתנה מחלקתי מסוג Counter בשם lifeCounter

• - לערך הנ"ל יש פונקציית increment,decrement האחריות להורדת הערך\הוספה וכל הנגזרות של הפעולה הזו.

3) על מנת לממש את כל ההתנהגויות יצרנו לכל אחת מהן מחלקה אשר עושה override לשיטת onCollision כל אחת יש לה אסטרטגיה משלה עם מימוש ייחודי בהתאם לצרכים. יצרנו מחלקות Heart, Puck, SecondPaddle כדי להגדיר את ההתנהגויות של כל אחד מהאובייקטים הנוספים למשחק. Puck יורשת מBall, מאחר והם מתנהגים כמעט אותו הדבר אבל לPuck יש עוד כמה התנהגויות רצויות. החלטנו להשתמש בירושה מאחר וPuck מקיים שהוא is-a Ball. באופן דומה גם SecondPaddle יורש מPaddle

כמו כן, יצרנו 'מפעל' של אסטרטגיות במפעל הזה מייצרים עבור המשחק אסטרטגיות בהתאם ללוגיקה שהנחתם אותנו. במימוש המשחק, לכל לבנה מתקבלת אסטרטגיה מהמפעל שכמובן בהסבתרות של חצי יכולה להיות onCollision הרגיל. כמו כן, יש ממשק ייעודי להתנהגות המוגדרת עבור ההתנגשות.

4) את המימוש של ההתנהגות הכפולה מימשנו בתוך המפעל, המפעל בעל משתנה בשם

STRATEGIES\_MAX במשתנה זה אנו הגדרנו מה המקסימום אסטרטגיות שיכולות להתקבל.

על מנת לפשט את הלוגיקה המתרחשת בפונקציות קודם כל הגרלנו רשימה של מספרים בין 0ל5 שמשמעותן היא איזו מבין האסטרטגיות המיוחדות התקבלה, את הרשימה של האינטגרים העברנו לפונקציה שיצרה מערך של אסטרטגיות מתאימות בהתאם ל'קידוד המספרי' כלל הלוגיקה מתרחשת בMakeStrategy אשר מקבל מס' קידוד ומחזיר אסטרטגיה רלוונטית.

כמובן שההגבלה שלנו היא מאוד פשוטה וכל מה שצריך לשנות על מנת לקבל מס' התנהגויות גדול יותר היא להגדיל את המשתנה מקס סטרטגיס ובכך הקוד שלנו גמיש בהחלפת שורה אחת בלבד.

5) הוספנו גטרים וסטרים .