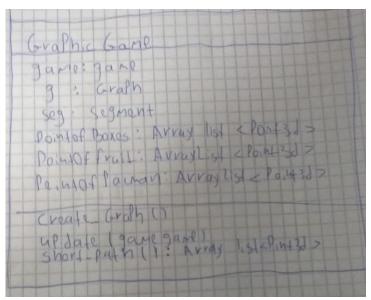
ברצוננו לבנות מערכת אוטומטית המשחקת מול השרת ומטרתה להשיג כמה שיותר נקודות.

בבסיס, המערכת שלנו תתבסס על גרף המורכב מצמתים ונקודות כאשר המטרה הראשית היא לעבור דרך **כל** צומת המוגדרת כ"פרי" .

את הגרף נבנה בכל איטרציה במשחק שכן בכל רגע נתון המשחק משתנה ולכן נצטרך לעדכן את הגרף.

את הגרף הזה נכתוב במחלקת GraphicGame שהבנאי שלה יקבל משחק ומהמשחק הזה ניצור גרף. גרף.

בנוסף מחלקה זו תכיל פונקציה בשם update שתעדכן את הגרף בהתאם לכל שינוי במשחק. המחלקה תכיל גם פונקציה בשם shortPath שתחזיר את המסלול הקצר ביותר לפרי או פקמן. פונקיה נוספת היא startingPoint אשר תקבע מה תהיה נקודת ההתחלה של הגרף.



דיאגרמת מחלקות של המחלקה.

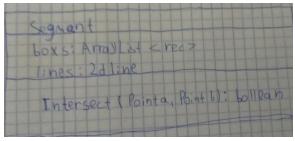
פירוט לגבי שלושת הפונקציות הללו נמצא בהמשך המסמך.

על המערכת להתמודד עם מספר פרמטרים נוספים:

ראשית נרצה להתמודד על הבעיה שהשחקן שלנו לא יעבור דרך נקודה שנמצאת בתוך קופסא או מחוץ למסגרת של המשחק.

או במילים אחרות נרצה ליצור קשתות בגרף שלנו כך שהם שלא נחתכות עם אחד מהמלבנים. לצורך כך נבנה מחלקה חדשה בשם segment.

מחלקה זו תחזיק בתוכה מערך של קופסאות ומערך של קווים. הבנאי של המחלקה יקבל מערך של קופסאות ומערך של קווים. הבנאי של המחלקה יקבל מערך של קופסאות ויהפוך כל קופסא לארבעה קווים- למעלה, ימינה, למטה ושמאלה . במחלקה זאת תהיה פונקציה אחת בשם Intersect אשר תקבל לתוכה שתי נקודות (המייצגות קו) ותבדוק האם הקו שהתקבל נחתך על ידי אחד מהקווים שבמחלקה.



דיאגרמת מחלקות של המחלקה

עכשיו נוכל להתחיל ליצור את הגרף. נתחיל מהוספת הנקודות הנמצאות מרחק קטן מאוד (אפסילון) מכל פינה של מלבן שכזה ונוסיף אותם לגרף. לאחר מכן נוסיף לגרף את הפירות שכרגע קיימים במשחק , לגבי הפקמנים נוסיף שאלה , האם הפקמן נמצא כרגע במקום שאנחנו יכולים להגיע אליו ? אם כן נוסיף גם אותו.

לאחר שהוספנו את כל הנקודות האלו (בתור צמתים) לגרף נרצה להוסיף לגרף את הצלעות, עכשיו נשתמש במחלקה שבנינו ועבור כל נקודה נעבור על כל שאר הנקודות בגרף ונשאל אם יש קו ישיר בניהם אם כן נוסיף אותם לגרף.

• נרצה להחליט היכן למקם את השחקן שלנו , על מנת לפתור שאלה זאת נסתכל על הגרף שיצרנו.

נקודת ההתחלה שלנו תמיד תהיה על פרי או פקמן ונעשה זאת על פי האלגוריתם הבא: עבור כל צומת נבדוק לאיזה צומת יש הכי הרבה שכנים שהם פקמנים .

אם מצאנו פקמן אז נבחר את המיקום שלנו על הפקמן הזה.

אם זה פרי, ניקח את הפרי שסכום הצלעות אל השכנים הוא המינמלי . (מקרה זה גם במידה ואין פקמנים בכלל) .

- על מנת למצוא את המסלול הקצר ביותר לכל נקודה נפעיל על הגרף את אלגוריתם
 דיקסטרה. כעת יש לנו את המרחק הקצר ביותר לכל נקודה בגרף. נעבור רק על הנקודות
 שהם או פקמן או פרי , נמצא את המרחק המינמלי ונחזיר את המסלול אל נקודה זאת.
 - פרמטר נוסף שנרצה להוסיף למערכת הוא התחמקות מרוחות.
- דבר ראשון נגדיר את אזור הסכנה שלנו הוא 30 מ', כלומר נבדוק בכל איטרציה אם השחקן שלנו נמצא ברדיוס של 30 מ' מרוח. אם כן אנחנו במצב מסוכן ונרצה להתחמק מהרוח. על מנת לעשות זאת נבנה פונקציה בשם runaway שתקבל שתי זוויות, זווית אחת היא הזווית שלנו ביחס לרוח והזווית השנייה היא הזווית שבה השחקן שלנו מתקדם כרגע במשחק. בפונקציה נבדוק אם שתי הזוויות נמצאות באותו רביע, אם כן נשנה את הזווית של הפקמן שלנו לזווית המנוגדת לזווית אליה הרוח מתקדמת לכיוונינו .

: מצורף פסאדו-קוד עם הסברים

