

משימת פיתוח - ספריית Arcade

עבודה עם קבצים ופיצ'רים מתקדמים

הקדמה

במהלך המפגשים הקודמים למדנו את יסודות פיתוח משחק וידאו, ובמפגש האחרון השלמנו את הפרויקט על בסיס כל הנושאים שנלמדו, עד **להגעה למשחק פועל במלואו**. כעת אנו עוברים לשלב הבא, שבו נוסיף למשחק פיצ'רים מתקדמים.

במשימה זו נעבוד עם מספר קבצים, ונלמד כיצד לאפשר למשחק **לקרוא נתונים מקבצים חיצוניים**. שלב זה יקרב אותנו לאופן שבו משחקים אמיתיים בנויים ומנוהלים בפועל.

בנוסף, במידה ויישאר זמן, ניישם פיצ'רים נוספים שיהפכו את המשחק לייחודי ומיוחד - רעיונות שאינם קיימים במשחק המקורי, ומאפשרים לכם להביא לידי ביטוי יצירתיות וחשיבה עצמאית.

תרגיל חימום

מפת המשחק מוגדרת כיום בראש הקובץ הראשי של המשחק שלכם Pac-Man.py (או כפי שקראתם לקובץ הראשי של המשחק שלכם).

כעת אתם מתבקשים להעביר את הגדרות המפה לתוך קובץ **טקסט** חיצוני מסוג **.txt**. ואז לטעון את המפה מתוך קובץ זה, ע"י שימוש בפקודות לעבודה עם קבצים שלמדתם בשיעור.

המטרה היא לטעון את מפת המשחק מהקובץ, ולוודא שהתוכנית מצליחה לקרוא את הנתונים בצורה תקינה.

לאחר ביצוע הטעינה, המשחק **אמור לרוץ בצורה מושלמת, ללא תקלות וללא שינוי בהתנהגות שלו**.

משימה

טעינת משתנים כלליים

כעת נמשיך ונרחיב את הפרויקט עוד יותר, ונטען משתנים מתוך קבצי טקסט חיצוניים (.txt).

לדוגמה, משתנים כמו:

- TILE_SIZE
- מהירות דמויות
- גובה ורוחב החלון

שימו  - **אין ציפייה שלכולכם יהיו בדיוק אותם משתנים/קבצים.**

כל תלמיד מימש את הפרויקט בצורה יצירתית ושונה, ולכן גם קבוצת המשתנים שכל אחד יבחר לטעון - יכולה להיות שונה, וזה לגמרי בסדר.

טעינת אקסטרה משתנים!

האם יש משתנים נוספים שניתן לטעון מקובץ?

!! חשוב: כל ערך קבוע שמשפיע על התנהגות המשחק, איזון, או עיצוב - הוא מועמד טוב לטעינה מקובץ.

גישה זו הופכת את המשחק לגמיש יותר, קל יותר לשינוי, ומדמה עבודה על פרויקט אמיתי שבו נתונים מופרדים מהקוד עצמו.

דגשים

בפרויקט, כל ה-Sprites ומכניקות המשחק חייבים לפעול בדיוק כפי שפעלו לפני טעינת המשתנים מקבצים חיצוניים. התנהגות המשחק אינה אמורה להשתנות, ואם היא כן משתנה – **יש לעבור על העבודה שלכם ולבדוק אותה מחדש.**

בונוס

הוספת פיצ'רים יצירתיים

פיצ'ר הוא תוספת או יכולת חדשה שמרחיבה את המשחק מעבר לבסיס הקיים.

להלן מספר רעיונות לפיצ'רים שניתן להוסיף למשחק - פירוט בעמודים הבאים:

1. תפוחים ורוחת
2. מפתחות ודלתות נעולות
3. אריחי טלפורטציה
4. אזורי בטיחות (Safe Zones)

הערה: הרעיונות נועדו להכווין, **אך לא להגביל** – אנחנו משאירים לכם מקום להיות יצירתיים, להפעיל דמיון, ולחקור

כיצד ניתן לממש את הרעיונות שלכם באמצעות Arcade.

💡 **אין פתרון אחד נכון** – כל מימוש אישי נותן למשחק טאצ' ייחודי משלכם!

בדיקה

בדקו את העבודה שלכם היטב! הריצו את המשחק מספר פעמים ונסו לבצע פעולות שונות בכל הרצה.

חשבו האם כסיתם את כל המקרים האפשריים, ומה קורה במצבים קיצוניים או לא צפויים.

ודאו שכל פיצ'ר שפיתחתם עובד בדיוק כפי שתכננתם, ושאינו פוגע בהתנהגות הכללית של המשחק. בדיקות חוזרות הן

חלק בלתי נפרד מפיתוח תוכנה איכותי!

בהצלחה!



1. תפוחים ורוחות

רעיון כללי:

בגרסה שלנו של פקמן, הרוחות מפחדות מתפוחים.
כאשר השחקן אוכל תפוח, הרוחות הופכות לפגיעות וניתן לאכול אותן לזמן מוגבל.

שאלות מנחות לשלבים:

1. איזה אובייקט חדש צריך להוסיף למשחק כדי לייצג תפוח?
כיצד תבדילו אותו ויזואלית משאר האובייקטים?
2. כאשר השחקן מתנגש בתפוח:
איך תסמנו במשחק שמצב כוח לשחקן (Power Mode) הופעל?
3. בזמן שמצב הכוח פעיל:
איך השחקן יבין שהרוחות ניתנות לאכילה?
האם תשנו את צבע הרוחות? אולי גם את צבע פקמן?
האם תרצו להוסיף סאונד מיוחד שמסמן שינוי מצב?
4. אם השחקן מתנגש ברוח בזמן שמצב הכוח פעיל:
מה צריך לקרות לרוח?



2. מפתחות ודלתות נעולות

רעיון כללי:

חלקים מסוימים במפה חסומים על ידי דלתות נעולות, וניתן לעבור דרכן רק לאחר איסוף מפתח.

שאלות מנחות לשלבים

1. איזה אובייקט חדש ייצג מפתח?
כיצד תגרמו לו להיראות שונה משאר האובייקטים?
2. כיצד תייצגו דלת נעולה?
האם היא תתנהג כמו קיר רגיל?
3. כאשר השחקן אוסף מפתח:
איך תשמרו בזיכרון המשחק שהמפתח נאסף?
4. לאחר שהמפתח נאסף:
מה יקרה לדלתות הנעולות?
האם תסירו אותן? או רק תבטלו את ההתנגשות שלהן?
5. כיצד השחקן יבין שנפתח לו אזור חדש במפה?



3. אריחי טלפורטציה

רעיון כללי:

אריחי טלפורטציה מעבירים את השחקן מיידית ממקום אחד במפה למקום אחר.

שאלות מנחות לשלבים:

1. כיצד ייראה אריח טלפורט?
איך השחקן ידע שמדובר באריח מיוחד?
2. כיצד תגדירו זוגות של טלפורטים (כניסה ויציאה)?
3. כאשר השחקן "דורך" על אריח טלפורט:
אילו ערכים צריך לשנות כדי להעביר אותו מיידית למיקום חדש?
4. איך תמנעו מצב שבו השחקן "קופץ" הלוך וחזור בלי שליטה?

4. אזורי בטיחות (Safe Zones)

רעיון כללי:

במפה קיימים אזורי מיוחדים שבהם השחקן מוגן מפני הרוחות.

שאלות מנחות לשלבים:

1. כיצד תייצגו אזור בטוח בצורה ויזואלית ברורה?
האם ע"י צבע שונה, דפוס מיוחד, סימון על האריחים?
2. כיצד תזהו שהשחקן נמצא בתוך אזור בטוח?
האם תבדקו התנגשות עם SpriteList ייעודי?
3. מה קורה כאשר השחקן נמצא באזור הבטוח?
האם הרוח יכולה להיכנס לאזור זה?
האם פשוט לא מתבצעת פגיעה?
4. מה קורה כאשר השחקן יוצא מהאזור הבטוח?
כיצד תחזירו את התנהגות המשחק למצב הרגיל?

הערות חשובות:

- כל הפיצ'רים **אופציונליים ובלתי תלויים** זה בזה.
- ניתן לממש אותם בכל סדר.
- לוגיקה ברורה וקריאה חשובה יותר ממורכבות.
- פרטים קטנים לכאורה: צבעים, סאונד, תזמון – עושים הבדל גדול בחוויית המשחק!