

פרוייקט בהנדסת תוכנה ואבטחת איכות תש"ע

סנייק (SNAKE)

המערכת:

הדמיית המשחק סנייק.
משחק הסנייק הומצא באמצע שנות השבעים וצבר פופולריות כאשר הוסף לטלפונים הסלולרים בשנות התשעים.
במשחק שולט השחקן על נחש ארוך המשוטט באזור מבודד, אוסף אוכל (או פריטים דומים) ומנסה להימנע מלהתנגש בזנבו או בקירות המקיפים את איזור המשחק. הנחש נמצא בתנועה מתמדת וכל עצם (שהוא לא רעל וכד') שהנחש אוכל, מאריך את גוף הנחש. (ויקיפדיה)
מטרת המשחק – לצבור את הניקוד הגבוה ביותר ולהאריך את גוף הנחש ע"י אכילת עצמים ללא התנגשות בגוף הנחש ובקירות המשחק.

מרכיבי המערכת:

סנייק:

נקודה, מרובע או כל עצם בצורה גיאומטרית.
בתחילת כל משחק סנייק מתחיל ממקום קבוע מראש על גבי לוח המשחק. סנייק נע על גבי הלוח בין המכשולים במטרה לאכול את כל הפירות על גבי לוח המשחק. התנגשות בקירות המשחק או בגוף הסנייק שווה להורדת פסילה
כל משחק מתחיל כאשר לשחקן יש 3 פסילות ו- 0 נקודות.

פירות:

תפוח – מופיע על המסך במיקום אקראי וישאר על גבי המסך עד לאכילתו (לא יופיע תפוח נוסף עד לאכילת התפוח הקיים על המסך). 5 שניות מרגע אכילת התפוח יופיע תפוח חדש. ניקוד - 10
תוספת אורך לסנייק – 1 יח'

בננה – תופיע אקראית על המסך ותשאר על גביו עד לאכילתה (לא תופיע בננה חדשה עד לאכילת הבננה הקיימת). 10 שניות מרגע אכילת הבננה תופיע בננה חדשה. ניקוד - 15
תוספת אורך לסנייק – 1 יח'

אגס – יופיע באחת מפינות המסך עם תחילת המשחק ועד לאכילתו (לא יופיע אגס חדש עד לאכילת האגס הקיים). לאחר אכילתו יופיע בפינה אחרת, כך שלא יהיה מצב בו יופיע ברצף באותה פינה פעמיים. ניקוד – 20
תוספת אורך לסנייק – 1 יח'

עכבר:

נקודה, מרובע או כל עצם בצורה גיאומטרית, יופיע במיקום אקראי, וינוע על גבי מסך המשחק בצורה רנדומלית (העכבר לא יכול לעבור דרך מכשולים) עד לאכילתו. עכבר חדש יופיע דקה אחרי אכילת העבר הקודם (לא יופיע עכבר חדש עד לאכילת העכבר הקיים). אכילת העכבר תוסיף חיים (פסילה) לסנייק.
ניקוד – 30
תוספת אורך לסנייק – 2 יח'

שאלות:

השאלות במשחק צריכות להיות במבנה של שאלות אמריקאיות, כאשר לכל שאלה 4 תשובות אפשריות. השאלות יכתבו ע"י חברי הקבוצה בנושאי הקורס.
קיימות 3 רמות של שאלות – שאלה קלה (צבע לבן) – מזכה ב- 1 נק', שאלה בינונית (צבע צהוב) – מזכה ב- 2 נק', ושאלה קשה (צבע אדום) – מזכה ב- 3 נק'. מענה נכון על השאלה יזכה בניקוד השאלה. מענה שגוי יוריד ניקוד בהתאם לרמת הקושי – שאלה קלה 10- נק', שאלה בינונית 20- נק', שאלה קשה 30- נק'.
בכל שלב במשחק צריכות להיות 3 שאלות על גבי לוח המשחק במיקומים אקראיים (מיד לאחר אכילת שאלה תופיע שאלה חדשה במיקום רנדומלי).

קירות:

מהווים חסמים שלא ניתנים למעבר, מפגש בקיר פירושו פסילה. לא ניתן למקם אובייקטי ניקוד על קיר.

שאלות:

השאלות שיוצגו בעת אובייקט שאלה יטענו מתוך קובץ JSON שיסופק על ידי צוות הקורס. (אין להקים בסיס נתונים. יש להשתמש בקובץ JSON בלבד)
אין לערוך או לשנות את מבנה הקובץ (יש למלא אותו בשאלות).
כל שאלה תכיל 4 תשובות אפשריות כאשר רק אחת מהן היא תשובה נכונה.

סיום משחק:

המשחק יסתיים כאשר השחקן מיצה את כל הפסילות (חיים).

זרימת התהליך:

1. מסך פתיחה המאפשר התחלת משחק, צפייה בהסטורית משחקים (נתון לפרשנות אישית), אשף ניהול שאלות.
2. התחלת משחק, הזנת שם/כינוי השחקן.
3. הצגת לוח המשחק. על גבי לוח המשחק יוצג הניקוד הנוכחי, מספר הפסילות ושאלות ותשובות בעת אכילת אובייקט שאלה.
4. מהלך משחק. סנייק מתחיל באמצע המסך, אוכל את האובייקטים השונים תוך הימנעות ממכשולים ומאובייקטים המורידים ניקוד, פסילות ואורך גוף.
5. אשף ניהול שאלות. מסכי ניהול שיאפשרו עריכה, הוספה ומחיקת שאלות.
6. סיום משחק, כל הפסילות מוצג. ניקוד המשחק נשמר לפי שם המשתמש.

שמירת נתונים:

יש לחשוב על מבנה נכון של התוכנה ובחירה מתאימה של מבני נתונים לצורך שמירת הנתונים במערכת. תהיה מחלקה אחת מרכזית שתחזיק את נתוני הסטוריית המשחקים ונתוני השאלות בשם SysData. יש לחשוב על קשרים מתאימים בין המחלקות ויצירת האובייקטים המתאימים לצורך פיתוח המערכת על כלל הפונקציונליות המתוארת.

הנחיות נוספות:

- הפרוייקט ייכתב בשפת התכנות JAVA ובסביבת עבודה IntelliJ או Eclipse בלבד.
- ניהול התצורה יתבצע (version control) באמצעות GitHub.
- שמירת נתוני השאלות תתבצע באמצעות קובץ JSON עם סכמה קבועה. אין לשנות את סכמת האובייקטים בקובץ.
- הפרמטרים לבדיקה יכללו: יכולת הרצת הקוד בכל שלב בהגשה, מבנה נכון יעיל ודינאמי של התוכנית, פונקציונליות תקינה ומלאה, חלוקה נכונה של ארכיטקטורת התוכנה (מתודות, מחלקות) לפי MVC, שימוש ב Design Patterns, תיעוד ברור ומלא, ממשק נקי ונוח למשתמש, בדיקת מקרי קצה.
- תוספות והשקעה ייתרה יהוו בonus.
- בכל הגשה בכל אחת מהאיטרציות, תרגיל שלא ירוץ ולא יתקמפל הציון עליו יהיה 0.
- על אחריות הסטודנטים לוודא תקינות הקובץ על יותר ממחשב אחד טרם ההגשה.

פירוט בonus:

- יינתן בonus עד 15 נקודות לציון הפרוייקט הסופי על מימוש יצירתי ותוספות מעניינות למשחק.
- 3 פרוייקטים מצטיינים יקבלו בonus נוסף:
יינתן בonus ל-3 הפרוייקטים המצטיינים על פי תחרות שתיערך בסוף הסמסטר. 3 נק' לציון הסופי בקורס לזוכים במקום הראשון, 2 נק' לציון הסופי בקורס לזוכים במקום השני ו-1 נק' לציון הסופי בקורס לזוכים במקום השלישי.