

חזרה בנושא עץ בינרי

משחק שמאל ימין

מטרות

תרגול השימוש בפעולות של עץ בינרי
תרגול במידול
אלגוריתמיקה

רמת השאלה

בינונית.

הקדמה

המשחק "שמאל-ימין" מיועד ל-2 עד 100 שחקנים.

לוח המשחק הוא עץ בינרי שבצמתיו מספרים שלמים וחיוביים.

כל שחקן בתורו "מטייל" על לוח המשחק, החל משורש העץ. השחקן פונה שמאלה או ימינה לצומת הבא בעץ בהתאם להגרלת מספר (1 – ימינה, 0 – שמאלה), עד אשר הוא מגיע לעלה בעץ ושם הוא נעצר. תוך כדי ה"טיול" בעץ הוא צובר נקודות שהן סכום המספרים המופיעים בצמתים בהם ביקר (מהשורש ועד לעלה). אם במהלך ה"טיול" נדרש השחקן (על פי ההגרלה) לפנות לצומת שאינו קיים, הוא מפסיד 3 נקודות, מגריל מספר חדש וממשיך לשחק עד אשר הוא מגיע לעלה.

מה עליכם לעשות?

ממשו את הפעולה הבאה:

```
public static void playLeftRight(BinaryTree<Integer> tree, int n)
```

טענת כניסה: הפעולה מקבלת עץ בינרי `tree` המייצג את לוח המשחק "שמאל-ימין", ומספר שלם `n` המייצג את מספר השחקנים במשחק (השחקנים ממוספרים מ-1 עד `n`). יש להניח ש: $n \leq 100$ וגם $n \geq 2$.

טענת יציאה: הפעולה מדפיסה עבור כל שחקן את מספרו ואת הניקוד שצבר בסיום המשחק. הפלט יהיה ממוין בסדר יורד לפי הניקוד שצברו השחקנים: השחקן בעל הניקוד הגבוה ביותר יודפס ראשון ולבסוף יודפס השחקן בעל הניקוד הנמוך ביותר.

הנחיות מיוחדות

- שאלה זו בודקת את היכולת של התלמיד להשתמש בעץ בינרי.
- על התלמיד להבין שאת הניקוד של כל השחקנים יש לשמור באוסף כלשהו – נוח במערך.
- הדמיית המשחק יחסית פשוטה להבנה ומימוש – יש לבצע לולאת **for** שמתבצעת n פעמים (כמספר השחקנים) ובתוכה יש לבצע לולאת **while** שתפקידה לדמות את ה"טיול" של השחקן על העץ משורשו ועד הגעה לעלה. במהלך ה"טיול" צוברים את הערכים בצמתים בהם ביקר השחקן. ההחלטה האם לפנות ימינה או שמאלה בכל שלב תתבצע בצורה אקראית.
- החלק היותר קשה הוא בשלב הסופי – בו הפעולה צריכה להדפיס עבור כל שחקן את מספרו ואת הניקוד שצבר בסיום המשחק, בצורה ממוינת בסדר יורד: השחקן בעל הניקוד הגבוה ביותר יודפס ראשון ולבסוף יודפס השחקן בעל הניקוד הנמוך ביותר. בהנחה שהניקוד של כל השחקנים שמור במערך פשוט, שהאינדקסים שלו מציינים את מספרי השחקנים, תהיה בעיה להשתמש במיון (בועות למשל) על מנת לבצע משימה זו, מאחר והמיון יגרום לאובדן הקשר בין מספר השחקן (האינדקס במערך) לבין הניקוד שצבר. רמז לפתרון טוב יותר: השתמשו במחלקת עזר בשם `Player` שתגדיר שחקן ולה שתי תכונות: מספר השחקן ומספר הנקודות שצבר.
- רעיון אלגוריתמי אחר לפתרון בעיית הדפסת השחקנים בצורה ממוינת, הוא שלאחר שכל שחקן מסיים לשחק, יש להכניס אותו למבנה ממוין על פי הניקוד הסופי שצבר. מבנה כזה יכול להיות רשימה ממוינת או תור עדיפויות!
- הרחבות אפשריות - ניתן לבקש להוסיף לפלט עוד מידע כגון:
 - המסלול שעבר כל שחקן במהלך המשחק – סידרת מילים "שמאל"/"ימין" משורש העץ ועד ההגעה לעלה. רמז: השתמשו בתור!
 - מספר הפעמים שהשחקן הפסיד נקודות.

פתרון

מצורפים קבצי המקור הבאים:

שם הקובץ	תיאור
<code>LeftRightGame.java</code>	מחלקת בדיקה בה נמצא מימוש הפעולה <code>playLeftRight(...)</code>
<code>Player.java</code>	מחלקת עזר המגדירה שחקן, לו שתי תכונות: מספר מזהה ומספר הנקודות שצבר.

מקור השאלה

צוות הפיתוח.