

## תרגיל בית מספר 4

**נושא: עץ חיפוש בינארי גנרי ושחקני הוקי-קרח פולימורפיים**

**דדליין: יום ה', 27/12/2018, 23:59**

**הגשה ביחידים**

**בהצלחה רבה!**

### תיאור התרגיל

תרגיל זה מהווה הרחבה של תרגיל בית 3 בו מימשתם עולם מונחה עצמים של זירת הוקי-קרח. בתרגיל הנוכחי יתווספו שני מרכיבים לסימולציה הקיימת – ראשית, מבנה נתונים גנרי שיושמש עבור השחקנים, ושנית, טיפוס שחקנים נוספים והתנהגות פולימורפית בתוך הקבוצה. לפיכך, מטרת התרגיל היא כפולה – ראשית, תרגול תכנותי של **תבניות** במימוש מבנה נתונים גנרי, ושנית, המשך תכנון ומימוש של עץ הורשה נתון.

### חלק א': עץ חיפוש בינארי גנרי באמצעות קוד תבניות

בחלק זה עליכם לממש מבנה של "עץ חיפוש בינארי" באופן גנרי באמצעות **קוד תבנית**. "עץ חיפוש בינארי" הינו מבנה נתונים סדור לפי העיקרון הרקורסיבי "איברי תת-עץ שמאלי קטנים בערכם מן השורש, איברי תת-עץ ימני גדולים בערכם מן השורש". ההנחה היא כי יחס סדר מוגדר היטב על הערכים של האיברים המיועדים לעץ.

**המימוש שלכם נדרש לכלול את המרכיבים הבאים:**

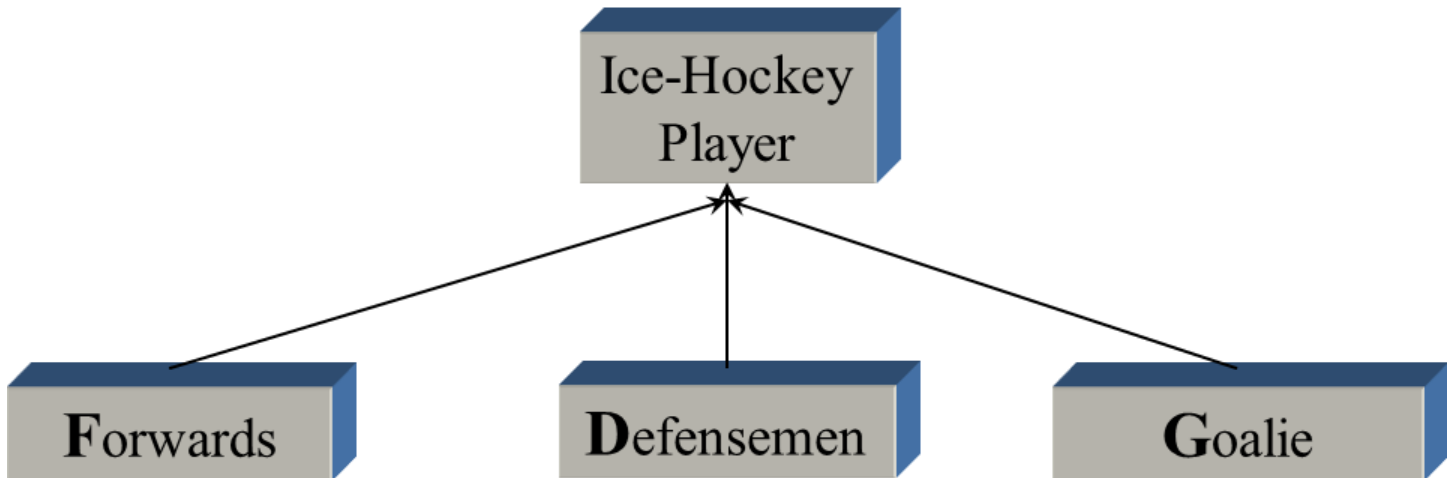
- פונקציית הכנסת איבר בשם `insert`; פונקציה זו תכניס לעץ איבר באופן השומר על העיקרון הרקורסיבי הנ"ל, תחת הנחה שאופרטור `<` ממומש עבור טיפוס הנתונים. לא תיתכנה כפילויות בעץ - הכנסת איבר שכבר קיים בעץ תגרום לדריסה.
- חיפוש איבר בעץ ע"פ ערכו והחזרת **מצביע** אליו אם וכאשר נמצא (מתודה בשם `search`)
- פונקציה בשם `min` המחזירה מצביע אל האיבר בעל הערך המינימלי בעץ
- פונקציה בשם `max` המחזירה מצביע אל האיבר בעל הערך המקסימלי בעץ
- פונקציית הסרת איבר בשם `remove`; בהינתן איבר, פונקציה זו תמחוק אותו מן הרשומות בהנחה שהינו קיים בעץ.
- פונקציית ניקוי העץ (מחיקת כל האיברים, כלומר, מעבר לעץ ריק) בשם `clear`
- פונקציית הדפסה לערוץ הפלט הסטנדרטי, אשר מדפיסה את כל איברי העץ באופן סדור, בשם `print`; סדר הדפסת האיברים הינו **סדר המיון העולה**. הפונקציה תסתמך על אופרטור הדפסה לפלט הסטנדרטי `<<` של האיבר.
- שאלתה על מספר האיברים המצויים בעץ.

מעבר לאפיון ולדרישות הללו, אתם חופשיים לתכנן ולממש כרצונכם מבנה נתונים זה, כל עוד נשמרת הפונקציונליות המתוארת. למשל, ניתן להוסיף מתודות עזר שיאפשרו מעבר בדיד על איברי העץ באמצעות מצביעים. שימו לב, כי אתם רשאים לדרוש תכונות מסוימות שעל האיברים לקיים כדי שיאוחסנו במבנה הגנרי שלכם. עליכם לפרט בתיעוד את כל הדרישות וההנחות להן אתם נזקקים.

**הנושא "מחלקות תבנית" בשפת C++ יילמד בהרצאה הקרובה, ביום ראשון 16/12.**

בחלק זה של התרגיל אתם תשדרגו את תכנית קבוצת ההוקי אותה מימשתם בתרגיל הקודם לכדי קבוצה בעלת שחקנים פולימורפיים הממלאים תפקידים שונים.

מעתה, שחקני הקבוצה בהן הסימולציה תתחשב הינם שלושה אפשריים: ('G') Goalie, ('D') Defensemen, ו-('F') Forwards. מבחינת תיאור אובייקטים, שלושתם שייכים לשחקני ההוקי (יורשים מאובייקט כללי "שחקן הוקי-קרח"). עץ הירושה המתאר משפחה זו נתון בסכימה הבאה:



עדכון השחקנים ימשיך להתרחש בכל פעימת זמן בדידה (דור), וישמור על אופיו הסתברותי, אך העדכון עצמו יהיה תלוי בסוג השחקן – כפי שיוסבר כאן בפירוט. תזוזה של כל שחקן בזירה תתרחש לפי משוואת התנועה המקורית, כלומר, עבור שחקן בעל אינדקס  $i$ , מיקומו בנקודת זמן  $t+1$  ייקבע ע"פ המשוואה הבאה:

$$\vec{\ell}_i(t+1) = \vec{\ell}_i(t) + \vec{v}_i(t) \quad (1)$$

ההבדל בין השחקנים השונים יבוא לידי ביטוי בעדכון רכיב המהירות. מהירותו של כל שחקן תמשיך להיות תלויה כמו בתרגיל הקודם בשלושה מרכיבים – בערך הקודם, במוביל הקבוצה ( $global\ best$ ),  $\hat{y}(t)$ , ובשיא האישי ( $personal\ best$ ),  $\vec{y}_i(t)$ :

$$\vec{v}_i(t+1) = \alpha \cdot \vec{v}_i(t) + \beta \cdot r_{1i}(t) \cdot (\vec{y}_i(t) - \vec{\ell}_i(t)) + \gamma \cdot r_{2i}(t) \cdot (\hat{y}(t) - \vec{\ell}_i(t)) \quad (2)$$

הערכים של הפרמטרים  $\{\alpha, \beta, \gamma\}$  ייקבעו ע"פ סוג השחקן:

- לשוער ("G") יהיו פרמטרים בעלי ערכים  $\{\alpha_G = 0.01, \beta_G = -0.1, \gamma_G = -0.25\}$
- לשחקן ההגנה ("D") יהיו פרמטרים בעלי ערכים  $\{\alpha_D = 0.05, \beta_D = 0.1, \gamma_D = 0\}$
- לשחקן ההתקפה ("F") יהיו פרמטרים בעלי ערכים  $\{\alpha_F = 0.25, \beta_F = 1.0, \gamma_F = 1.0\}$

# תכנות בשפת C++, סתיו 19-2018

## שינויים בהשוואה לתרגיל הבית הקודם

- עליכם לעשות שימוש במבנה הנתונים הגנרי אותו מימשתם בחלק א' של התרגיל בכדי לאחסן את שחקני הקבוצה באופן פולימורפי; כלומר, אובייקט הקבוצה מצופה להחזיק את השחקנים באמצעות מבנה זה. מיון השחקנים בעץ יהיה פשוט על-סמך האינדקס המספרי שלהם אשר נקבע ע"פ סדר הצגתם בקובץ הקלט בשעת איתחול הסימולציה.
- התכנית תקבל גם בתרגיל זה שני קבצי קלט – הראשון יתאר את זירת ההוקי ויהיה בעל פורמט זהה לתרגיל הקודם, והשני יכלול הן את פריסת השחקנים ההתחלתית והן את המהירויות ההתחלתיות שלהם, אך יכלול שינוי בפורמט כדי לאפיין את סוג השחקן. הקובץ יכיל בשורה הראשונה את מספר השחקנים בקבוצה, ובשורות לאחר מכן את סוגם ומצבם ההתחלתי. כל שורה תתחיל בתו המגדיר את טיפוס השחקן (התווים האפשריים הינם '{G','D','F'}), ולאחר מכן תכיל את מיקומו ההתחלתי של השחקן המתאים (באמצעות 2 קואורדינטות) ואת מהירותו ההתחלתית של אותו השחקן (באמצעות 2 רכיבי מהירות), מופרדים ברווחים. להלן דוגמה לקובץ קלט חוקי של איתחול השחקנים:

```
6
G 3 15 0.01 0.01
D 12 7.5 0.35 0.25
D 12 22.5 0.2 -0.2
F 21 7.5 0.3 0.1
F 21 22.5 0.1 -0.1
F 30 15 0.6 0.05
```

שימו לב כי האינדקס של כל שחקן (המהווה את המפתח שלו בעץ) הינו ביחס לקבוצה כולה ולא ביחס לסוג השחקן. למשל, בדוגמה שניתנה, יש 6 שחקנים עם אינדקסים עוקבים [1..6].

בסיום מעברי הזמן, על התכנית להחזיר כפלט את מספר צעדי הזמן של הסימולציה וכן את מיקומי השחקנים של צעד הזמן האחרון באמצעות ערוץ הפלט הסטנדרטי. על נתונים אלו להיות מודפסים, כמו בתרגיל בית 3, בסדר הרכבים שניתן בקובץ הקלט (הרכב הראשון בקובץ הקלט יהיה הראשון בפלט וכו'). למשל, כך עשוי להיראות הפלט עבור הדוגמא שלעיל –

```
96717
G 3.01 13.96
D 10.45 7.51
D 13.87 20.9
F 58.18 14.84
F 57.99 17.1
F 58.32 14.44
```

## הנחות עבודה

הנחות העבודה מהתרגיל הקודם לגבי מקרי קצה של מספר הדורות, טיפול בתנאי השפה, וגודל הקבוצה תקפות גם בתרגיל זה. כמו כן, הטיפול בקלט בלתי-חוקי נדרש להיות זהה לתרגיל הקודם.

## דגשים

- אין אפשרות להשתמש בספריית STL בתרגיל זה.
- עליכם לוודא כי התכנית עוברת קומפילציית g++ התואמת את הקומפיילר שעל שרת המכללה ללא כל שגיאות או אזהרות כלשהן, ורצה בהצלחה.
- עליכם לתעד את הקוד באמצעות הערות המתארות בקצרה את הפונקציות השונות.
- יש להריץ את הבודק האוטומטי על שרת החוג בטרם ההגשה בכדי לוודא תאימות ונכונות של ההגשה: התחברו לשרת החוג והריצו **hwcheck** על הארכיב שלכם, או לחילופין העלו את הארכיב תוך שימוש בפרוטוקול HTML

בקישור <http://cs.telhai.ac.il/homework>

- עליכם להגיש במערכת Moodle קובץ ארכיב מטיפוס zip בלבד, ששמו כולל את קוד הקורס ('11'), שם התרגיל ('ex4') ותעודת הזהות של הסטודנט/ית המגיש/ה, מופרדים בקו תחתי בפורמט הבא: **11\_ex4\_studID.zip**
- על ארכיב zip זה להכיל את כל קבצי המקור (ממשק/מימוש) הנדרשים לקומפילציה, והוא רשאי להכיל תיעוד טקסטואלי; מבחינת טיפוס קבצים, עליו לכלול רק קבצים עם סיומות **\*.cpp \*.h \*.txt**
- לדוגמא: על סטודנט/ית בעל/ת מספר זיהוי 012345678 להגיש ארכיב בשם 11\_ex4\_012345678.zip הכולל את כל קבצי המקור של הפרוייקט, ללא תיקיות כלשהן, ורשאי להכיל קובץ טקסטואלי לתיעוד.

אי-הקפדה על ההנחיות, כולל פורמט ההגשה הדיגיטלי, תגרור הורדה בציון התרגיל.  
לא תתקבלנה הגשות באיחור!