

[פרק 11 – templates]

{רשימת המשימות לביצוע}

נושא תרגול זה הוא templates: פונקציות מבוססות template'ים, מה קורה מאחורי הקלעים מבחינת תהליך הקומפילציה והלינקר כאשר כותבים קוד מבוסס templates, ונראה גם מחלקה template'ית.

בדף זה מתוארים הסעיפים השונים, ומומלץ לנסות לקודד אותם לבד לפני שמקשיבים להקלטה. תרגול זה מסתמך על פתרון תרגול 8 (פולימורפיזם), ולא על תרגול 10, כדי לאפשר גמישות וקפיצה על הפרקים 9 ו-10. להורדת הקוד של התרגול הקודם (פולימורפיזם) שממנו תוכלו להמשיך לקדד, לחצו [באן <<](#)

[רשימת המשימות]

[3] פונקציית template – מימוש ומוטיבציה

{✓} כתבו פונקציה המקבלת מערך וגודלו ותחזיר את הערך הגדול ביותר במערך. השוואה בין האיברים תשתמש באופרטור <

[4] פונקציית template – שימוש במחלקה

{✓} השתמשו בפונקציה זו במחלקה Team על מנת להחזיר את השחקן שקלע הכי הרבה נקודות

[5] מדוע אין הפרדה בין H ל- CPP?

{✓} נסו להפריד את המימוש של פונקציות ה- template שכתבתם לקובץ CPP נפרד ובדקו אילו שגיאות אתם מקבלים. נסו להבין מדוע.

[6] כתיבת מחלקת template

- {✓} כתבו את מחלקת ה- template ששמה Array שתחזיק מידע על מערך שאיבריו מטיפוס **מצביעים ל-T**: מצביע לתחילת המערך, גודל לוגי וגודל פיזי
- {✓} הוסיפו קונסטרקטור המקבל את הגודל הפיזי ההתחלתי ומקצה את המערך
- {✓} העמיסו את האופרטור += אשר יקבל מצביע לאיבר חדש ויוסיף הצבעה אליו. במידה ואין מקום, יש להגדיל את המערך פי 2
- {✓} העמיסו את האופרטור << אשר ידפיס כמה מקום יש במערך, כמה איברים בשימוש ואת ערך האיברים המוצבעים
- {✓} כתבו מתודת find המקבלת אובייקט ובודקת האם קיים איבר שערכיו זהים. בדיקת השיוויון תהיה באמצעות אופרטור ==. במידה וקיים אובייקט כזה במערך יוחזר המצביע אליו, אחרת יוחזר nullptr.
- {✓} ממשו copy c'tor, move c'tor, operator=, move operator=, d'tor – כדי שהמחלקה תהייה שלמה מבחינה תכנותית.
- {✓} הוסיפו והעמיסו את האופרטור [] במחלקה כך שיחזיר את האיבר במקום ה- i.
- {✓} העמיסו את האופרטור בשתי גרסאות: const ורגילה
- {✓} הוסיפו מתודת getSize שתחזיר את הגודל הלוגי של המערך.

[7] שימוש במחלקת template

{✓} במחלקה League החליפו את השימוש במערך מצביעים ל- Team במחלקת ה- Array שכתבתם. שימו לב למחוק כל קוד מיותר.

[8] סיכום מה ראינו בפרק

[עבודה נעימה!]