

המחלקה להנדסת מערכות תקשורת

אוני' בן גוריון בנגב

הנחיות הגשה:

1. הגשה הינה ביחידים בלבד.
2. יש להגיש את כל קבצי העבודה במערכת ה VPL ולוודא שאין שגיאות קומפילציה במערכת.
3. כל מחלקה תכלול קובץ header וקובץ cpp (מלבד מחלקת Array).
4. בתרגיל זה אתם לא רשאים להשתמש במחלקת vector.
5. הקפידו על הוראות ברורות וקריאות למשתמש לפני כל פעולת קלט.
6. יש להקפיד על שמות המחלקות והשיטות כפי שמצוין בעבודה (הקפידו על אותיות גדולות וקטנות).
7. הקפידו על תכנות נכון, זהו חלק מהציון:
 - a. שמות משתנים בעלי משמעות.
 - b. שימוש חוזר בקוד.
 - c. הקפדה על הזחות.
 - d. כתבו הערות שמסבירות את הקוד שלכם.
8. אין חובה להדפיס הדפסות מפונקציות destructor/constructor/set/get.
9. destructor מופעל אוטומטית בסוף ה scope של האובייקט ואין צורך לקרוא לו.
10. יש לשחרר את כל הזיכרונות שהוקצו דינמית ביציאה מכל פונקציה ובסיום התוכנית.
11. הקלטים יהיו מהטיפוסים החוקיים. ז"א בכל מקום נכניס את הטיפוס המתאים. אנחנו לא מתחייבים שהוא יהיה בטווח מסוים – אלא אם כן נאמר אחרת בשאלה עצמה.
12. בראש כל קובץ יש להוסיף בהערה את הטקסט הבא:

```
*/Assignment C++: 3
Author: Israel Israeli, ID: 01234567
/*
```

- כמובן שעליכם להחליף את הפרטים בפרטים שלכם.
13. נסו להבין מה רוצים לבדוק בתרגיל. לכל דף עבודה ישנה כותרת ובה מפורטים נושאי התרגילים לדף העבודה. אפשר להסיק ממנה מה מצופה בתרגילים עצמם. יש להפעיל שיקול דעת במקרה שיש סעיף שאינכם בטוחים מה לעשות בו. אם עדיין לא ברור, העלו שאלה בפורום הרלוונטי לתרגיל. נא לא לשלוח שאלות שאתם חושבים שהן שאלות שניתן לפתור בעזרת הגיון בריא.
 14. הארכות יינתנו רק במקרי מילואים, אבל ומחלה חריפה (שלא נדע) ובצירוף אישורים מתאימים.

תרגיל מס' 3

נושא התרגיל: פולימורפיזם, RTTI, exceptions

הערה חשובה: רשימת המתודות היא חלקית בלבד. היא מהווה את המינימום הנדרש ממכם. ייתכן ואף סביר להניח שתצטרכו להוסיף נוספות עבור המחלקות המתוארות. חישבו היטב מה באמת צריך בכל מחלקה. כמובן, שגם אם לא נאמר, לכל המחלקות חובה להוסיף Constructor ו- Destructor דיפולטיבים (אפילו אם הוא ריק). כמו כן עליכם להוסיף מתודות set ו- get לפי הצורך.

**** שימו לב - בעבודה זו עליכם לוודא כי ערכי הקלט נמצאים בטווח המתאים. הטיפול יעשה באמצעות exceptions (אין צורך לבדוק טיפוסים), במידה והוזן קלט לא תקין יש להדפיס הודעה מתאימה ולאפשר למשתמש להזין את הנתונים בשנית.**

1. מחלקת `Array<T>` – template

- שדות:
 - int – גודל המערך.
 - T* – מצביע לאיבר הראשון במערך.
- מתודות:
 - add – מוסיפה איבר בסוף המערך.
 - remove – מקבלת אינדקס של איבר ומוחקת אותו, על מנת לאפשר שחרור זיכרון תקין, הפונקציה תחזיר את הערך שנמחק T.
 - ** בסוף הפעולה נותר מערך בגודל size-1.**
 - getSize – מחזירה את גודל המערך.
- בנאים:
 - בנאי ברירת מחדל.
- אופרטורים:
 - << - הדפסת כל האיברים במערך.
 - [] – array index operator.

2. מחלקת `Shape` –

- שדות:
 - Char* – צבע הצורה (מחרוזת דינאמית), הצבעים האפשריים הם blue, red, green.
 - int - עובי צורה.
- מתודות אבסטרקטיות:
 - getPerimeter – מחזירה את ההיקף double.
 - getArea – מחזירה את השטח double.
- בנאים:
 - בנאי המקבל את כל הפרמטרים.

3. מחלקת `Square` –

מחלקה זו יורשת מ Shape.

- שדות:
 - double – אורך צלע הריבוע.
- מתודות:
 - draw – מציגה למסך ריבוע של כוכביות בהתאם לאורך הצלע (עבור מספר לא עגול נעגל כלפי מטה).
 - ממשו את המתודות האבסטרקטיות שהוגדרו ב Shape.

4. מחלקת – Circle:

מחלקה זו יורשת מ Shape.

- שדות:
 - int – רדיוס.
- מתודות:
 - ממשו את המתודות האבסטרקטיות שהוגדרו ב Shape.

5. מחלקת – OrthogonalTriangle:

משולש ישר זווית ושווה-שוקיים. מחלקה זו יורשת מ Shape.

- שדות:
 - double – אורך השוקיים.
- מתודות:
 - draw – מציגה למסך משולש ישר זווית של כוכביות בהתאם לאורכי השוקיים.
 - ממשו את המתודות האבסטרקטיות שהוגדרו ב Shape.

תוכנית main:

כתבו תוכנית היוצרת אובייקט מסוג Array של מצביעים ל Shape ומציגה תפריט למשתמש:
(1) הוספת צורה לסוף המערך. בבחירה זו המשתמש יתבקש לבחור את סוג הצורה ולהזין את הנתונים שלה. דוגמה להוספת צורה:

```
Choose 1 for Square, 2 for Circle, 3 for Triangle: 3
Enter Triangle's color: green
Enter Triangle's width: 5
Enter Triangle's side: 5
```

(2) מחיקת צורה לפי אינדקס.

(3) הדפסת פרטי הצורה לפי אינדקס. דוגמה להדפסת פרטי צורה:

```
OrthogonalTriangle details:
color=green, width=5, side=5
area= 12.5, perimeter=17.0711
*
* *
* * *
* * * *
* * * * *
```

**שימו לב כי יש להפעיל את המתודה draw עבור משולש וריבוע.

**שימו לב כי מתודת draw תוגדר אך ורק במחלקות Square ו OrthogonalTriangle (RTTI).

(4) יציאה מהתוכנית.