

## פתרון פרק שני מסלול תכנות מונחה עצמים בשפת C#

### בגרות במדעי המחשב ב', קיץ תשס"ז 2007

21. שאלה העוסקת בניהול אתר "הסקר השבועי", בו מתפרסם בכל שבוע סקר חדש עם שאלה אחת, לה 4 תשובות אפשריות, מתוכן ניתן לבחור אחת. משתתף יכול לענות רק על שאלת הסקר האחרון שפורסם. האתר שומר את 50 הסקרים האחרונים שנערכו, כולל הסקר השבועי החדש.

א. הגדרת המחלקות הנדרשות למימוש אתר "הסקר השבועי". עבור כל מחלקה יש לכתוב מה מייצגת, לכתוב לה כותרת, להגדיר תכונות ולכתוב תיעוד לכל תכונה.

ראשית נגדיר את המחלקות הנדרשות:

שם מחלקה	תיאור מחלקה
Survey	מחלקה המייצגת סקר שבועי
WeeklySurveySite	מחלקה המייצגת את אתר "הסקר השבועי"

נכתוב כותרת עבור שתי המחלקות ונגדיר עבורן תכונות עם תיעוד:

#### תכונות המחלקה Survey

עבור סקר יש לשמור את הנתונים הבאים: תאריך פרסום, השאלה הנשאלת, התשובות האפשריות לשאלה הנשאלת ומספר המשתתפים שבחרו בכל תשובה אפשרית. בהתאם לכך נגדיר את תכונות המחלקה המייצגת סקר שבועי:

class Survey	
<code>private DateTime publicationDate</code>	// תאריך פרסום הסקר בפורמט DD/MM/YYYY
<code>private string question</code>	// השאלה הנשאלת בסקר
<code>private string[] answers = new string[4]</code>	// מערך מחרוזות המייצגות 4 תשובות אפשריות לשאלה הנשאלת בסקר
<code>private int[] answersChoosersAmount = new int[4]</code>	// מערך מספרים המייצגים את כמות המשתתפים שבחרו בכל אחת מבין 4 התשובות האפשריות

#### תכונות המחלקה WeeklySurveySite

ניהול האתר מתבסס על שמירת 50 הסקרים האחרונים שנערכו. הסקרים יישמרו במערך, בגודל עד 50, של אובייקטים מטיפוס המחלקה סקר Survey.

לתמיכה במימוש פעולות, הקשורות בסקר האחרון שפורסם, תוגדר תכונה, שתציין **מיקום הסקר השבועי החדש** במערך הסקרים. יש לשים לב, שהוספת סקר חדש לאתר תתבצע גם כאשר ישנם כבר 50 סקרים באתר, ומערך הסקרים מלא עד המקום האחרון. במקרה כזה, סקרים חדשים ידרסו סקרים ישנים מתחילת המערך.

לתמיכה במימוש פעולות, הכרוכות במעבר כולל על מערך הסקרים, תוגדר תכונה, שתציין את **מספר הסקרים הקיימים בפועל באתר**. תכונה זו רלוונטית כל עוד מערך הסקרים לא התמלא עד תום, שכן אז מספר הסקרים בפועל קטן מ-50, ולא יידרש מעבר על תאים שטרם אותחלו באובייקטים של סקר. ברגע שהמערך מתמלא ב-50 סקרים, גם אם יתווספו סקרים חדשים, אשר ידרסו סקרים ישנים מתחילת המערך, עדיין יידרש לעבור על כל המערך, על 50 סקרים, ישנים וחדשים. לפיכך יש לשים לב, היות שמציין מיקום הסקר החדש יכול לחזור למקומות בתחילת המערך, לא ניתן להקיש ממנו על מספר הסקרים בפועל, ולכן נדרשת תכונה זו.

בהתאם לכך נגדיר את תכונות המחלקה המייצגת את אתר "הסקר השבועי":

<code>class WeeklySurveySite</code>		
<code>private Survey[] weeklySurveys = new Survey[50]</code>	<code>//</code>	מערך של עד 50 סקרים
<code>private int newSurveyPosition = -1</code>	<code>//</code>	מציין מיקום הסקר החדש במערך (יאותחל ל-1)
<code>private int numOfSurveys = 0</code>	<code>//</code>	מספר הסקרים בפועל באתר (יאותחל ל-0)

ב. הגדרת הפעולות הנדרשות לניהול אתר "הסקר השבועי". עבור כל פעולה יש לכתוב באיזו מחלקה יש להגדיר אותה, לכתוב עבורה כותרת ותיעוד. ניתן להניח שפעולות `get / set` מוגדרות עבור כל תכונה הנמצאת במחלקות המוגדרות בסעיף א.

ניהול אתר "הסקר השבועי" כולל את הפעולות הבאות:

- הוספת סקר חדש לאתר
- הדפסת שאלת הסקר החדש, וארבע התשובות האפשריות עליה
- קליטת תשובת משתתף לשאלת הסקר החדש, ועדכון מספר המשתתפים שבחרו בתשובה זו
- קליטת תאריך פרסום של סקר, ועבור שאלת הסקר שפורסמה בתאריך זה, הדפסת מספר המשתתפים שבחרו בכל אחת מהתשובות האפשריות, והתשובה שנבחרה על ידי המספר הגדול ביותר של משתתפים.

בהתאם לרשימה זו נגדיר את הפעולות, הכרוכות בניהול האתר, ונציין את מיקום הגדרתן:

מיקום הגדרה	פעולה
המחלקה WeeklySurveySite	הוספת סקר חדש לאתר
המחלקה WeeklySurveySite	הדפסת שאלת הסקר החדש
המחלקה WeeklySurveySite	הדפסת תשובות אפשריות לשאלת הסקר החדש
המחלקה WeeklySurveySite	קליטת תשובת משתתף לשאלת הסקר החדש ועדכון מספר המשתתפים שבחרו בתשובה זו
המחלקה Survey	עדכון מספר המשתתפים שבחרו בתשובה אפשרית לשאלת סקר
המחלקה WeeklySurveySite	איתור סקר לפי תאריך פרסום באתר
המחלקה WeeklySurveySite	הדפסת תוצאות סקר לפי תאריך פרסום באתר
המחלקה Survey	הדפסת תוצאות סקר: מספר הבוחרים בכל תשובה אפשרית והתשובה הנבחרת ביותר
המחלקה Survey	אחזור אינדקס תשובת הסקר הפופולארית

#### עקרונות למיקום הפעולות במחלקות

פעולות ניהול, המהוות את ממשק האתר למשתמש מודול התוכנה, תוגדרנה במחלקת ניהול האתר WeeklySurveySite. ביניהן ניתן למנות פעולות, הקשורות בסקר החדש שפורסם, או פעולות, הכרוכות בעיבוד כלשהו של כלל הסקרים באתר.

פעולות, הכרוכות בעיבוד נתוני התכונות הפנימיות של סקר, תוגדרנה במחלקה סקר Survey. פעולות אלו תהוונה עזר למימוש פעולות במחלקת ניהול האתר WeeklySurveySite. כך למשל, פעולת עדכון מספר המשתתפים, שבחרו בתשובה אפשרית לשאלת סקר חדש, משתמשת בפועל בפעולת - עדכון מספר הבוחרים בתשובה אפשרית - של המחלקה סקר. ובדומה, פעולת הדפסת תוצאות סקר לפי תאריך פרסום באתר משתמשת בפועל בפעולת - הדפסת תוצאות סקר - של המחלקה סקר.

פעולות לקליטת קלט ישיר מהמשתמש (מקלדת) -- תשובת משתתף לשאלת סקר חדש; תאריך פרסום של סקר שרוצים לצפות בתוצאותיו -- תמומשה במחלקת התוכנית הראשית. לפיכך, הפעולות לעדכון מספר הבוחרים בתשובת משתתף נבחרת והדפסת תוצאות סקר לפי תאריך פרסום באתר, המוגדרות במחלקת ניהול האתר WeeklySurveySite, מקבלות את הקלט הדרוש עבורן כפרמטרים המועברים מן התוכנית הראשית.

עתה נרשום כותרות ותיעוד עבור פעולות שתי המחלקות שהגדרנו בטבלה מעלה:

### פעולות המחלקה Survey

class Survey	
<code>public void UpdateAnswerChoosersAmount (int answerNumber)</code>	// פעולה המקבלת מספר של תשובה נבחרת
//	בין 1 ל-4 ומעדכנת את מספר המשתתפים שבחרו בתשובה זו
<code>private int PopularAnswerIndex ()</code>	// פעולה המחזירה מציין מיקום של
//	התשובה בעלת המספר הגדול ביותר של משתתפים שבחרו בה כתשובה לשאלת הסקר. במקרה שיש יותר מתשובה אחת עם
//	מספר מרבי זהה של משתתפים בוחרים יוחזר מציין מיקום התשובה הראשונה המתאימה
<code>public void PrintSurveyResults ()</code>	// פעולה המדפיסה את תוצאות הסקר:
//	מספר המשתתפים שבחרו בכל תשובה אפשרית לשאלת הסקר והתשובה שנבחרה ע"י המספר הגדול ביותר של משתתפים
//	במקרה שבו יש יותר מתשובה אחת עם מספר מרבי זהה של משתתפים בוחרים – תודפסנה כולן

### פעולות המחלקה WeeklySurveySite

class WeeklySurveySite	
<code>public void AddNewSurvey (Survey survey)</code>	// פעולה המוסיפה סקר חדש לאתר
//	הפעולה מקבלת אובייקט מטיפוס סקר, ומתבצעת באופן שבו יישמרו 50 הסקרים האחרונים באתר
<code>public void PrintNewSurveyQuestion ()</code>	// פעולה המדפיסה את שאלת הסקר החדש
<code>public void PrintNewSurveyPossibleAnswers ()</code>	// פעולה המדפיסה את 4 התשובות
//	האפשריות לשאלת הסקר
<code>public void UpdateNewSurveyAnswerChoosersAmount (int answerNumber)</code>	
//	פעולה המקבלת תשובת משתתף [מספר בין 1 ל-4] לשאלת הסקר החדש, ומעדכנת את מספר המשתתפים שבחרו בתשובה זו
//	הפעולה משתמשת בפעולות - עדכון מספר המשתתפים שבחרו בתשובה אפשרית - של המחלקה סקר
<code>private Survey LocateSurveyByDate (DateTime date)</code>	// פעולה המקבלת תאריך פרסום של סקר
//	הקיים באתר, מאתרת ומחזירה אובייקט
//	סקר המתאים לתאריך הפרסום שהתקבל
<code>public void PrintSurveyResults (DateTime surveyPublicationDate)</code>	
//	פעולה המקבלת תאריך פרסום של סקר הקיים באתר ומדפיסה את תוצאותיו: מספר המשתתפים שבחרו בכל תשובה
//	אפשרית לשאלת הסקר והתשובה בעלת, או התשובות בעלות, המספר הגדול ביותר של משתתפים בוחרים
//	הפעולה משתמשת בפעולות - הדפסת תוצאות סקר - של המחלקה סקר

ג. מימוש פעולה המחזירה את מספר הסקרים בהם השתתפו מעל 1,000 משתתפים.

הפעולה להחזרת מספר הסקרים עם יותר מ-1,000 משתתפים תמומש במחלקת ניהול האתר WeeklySurveySite, ותשתמש בפעולה להחזרת מספר המשתתפים בסקר, שתמומש במחלקה סקר Survey. הפעולה להחזרת מספר הסקרים עם יותר מ-1,000 משתתפים, שתיקרא Above1000ParticipantsSurveys לא תקבל דבר, תחזיר את מספר הסקרים, ותמומש במחלקה WeeklySurveySite באופן הבא:

```
public int Above1000ParticipantsSurveys ()
{
    int above1000Participants = 0;

    for (int i = 0; i < numOfSurveys; i++)
        if (weeklySurveys[i].NumberOfParticipants() > 1000)
            above1000Participants++;

    return above1000Participants;
}
```

ניתן לראות כי הפעולה משתמשת בפעולה NumberOfParticipants, שאינה מקבלת דבר, מחזירה את מספר המשתתפים בסקר, וממומשת במחלקה Survey באופן הבא:

```
public int NumberOfParticipants ()
{
    int numberOfParticipants = 0;

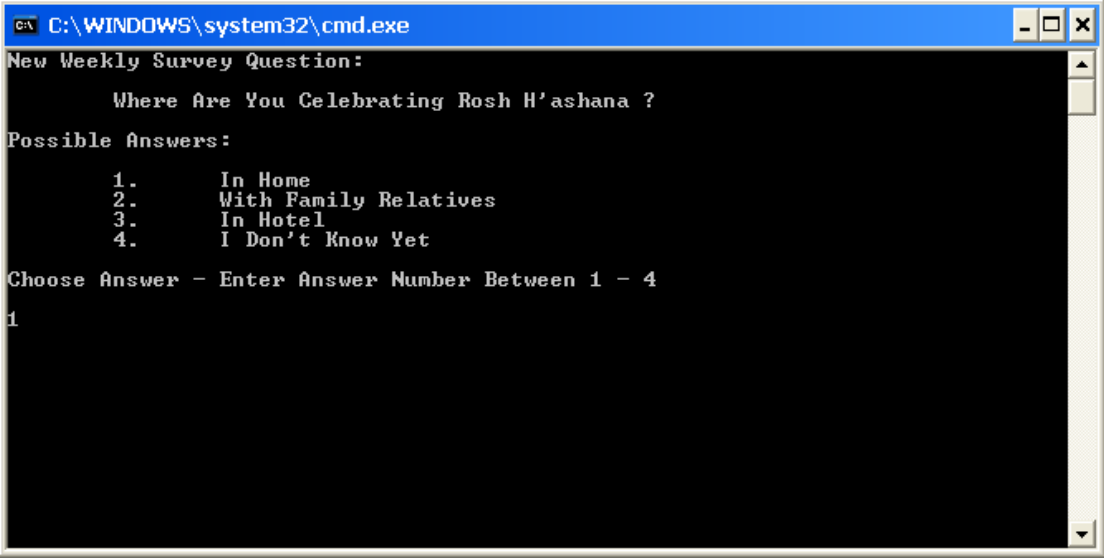
    for (int i = 0; i < 4; i++)
    {
        numberOfParticipants += this.answersChoosersAmount[i];
    }

    return numberOfParticipants;
}
```

פתרון מסלול תכנות מונחה עצמים שפת C#, שאלות 21 - 24, בגרות מדעי המחשב ב', קיץ תשס"ז 2007


להדגמת פעילות תוכנית, המתבססת על שתי המחלקות שהגדרנו, נאתחל מופע עצם עבור אתר הסקר השבועי, שבו פורסמו שני סקרים. הסקר הראשון פורסם בתאריך 15.09.2008 עם השאלה הרווחת לקראת תחילתה של שנה עברית חדשה: "היכן אתם חוגגים את ראש השנה?". הסקר השני פורסם בתאריך 22.09.2008 עם שאלה אודות תחביב ספורט מועדף: "איזה סוג של תחביב ספורט אתה אוהב?".

כך למשל תוצג באתר שאלת הסקר הראשון עם בקשה לבחירת תשובת משתתף בסקר:



```
C:\WINDOWS\system32\cmd.exe
New Weekly Survey Question:
      Where Are You Celebrating Rosh H'ashana ?
Possible Answers:
      1.      In Home
      2.      With Family Relatives
      3.      In Hotel
      4.      I Don't Know Yet
Choose Answer - Enter Answer Number Between 1 - 4
1
```

אם למשל נרצה להציג את תוצאות הסקר, שפורסם ב-22.09.2008 והשתתפו בו 5 משתתפים, נקבל את הפלט הבא:



```
C:\WINDOWS\system32\cmd.exe
Enter Survey Publication Date In Format DD/MM/YYYY
22/09/2008
Survey Question:
      Which Kind of Sport's Hobby Do you Like ?
Amount of Choosers Per Answer:
      Answer      Amount of Choosers
-----
Walking          1
Swimming         2
Running          1
Football Playing 1
Most Popular Answer:
      Swimming
Enter Survey Publication Date In Format DD/MM/YYYY
```

ניתן לראות בתוצאות הסקר את כמות המשתתפים שבחרו בכל תשובה אפשרית לשאלת הסקר ואת התשובה הרווחת לשאלה (שחייה).

**22.** שאלה העוסקת במחשוב מוזיאון "ישן וחדש", שבו מוצגים משלושה סוגים: ציור (Painting), פסל (Statue) וצילום (Photograph).

א. מימוש פעולה בונה במחלקה Painting, המשתמשת בפעולה הבונה המקבלת פרמטרים של המחלקה Exhibit:

```
public Painting(string creator, string name, int year, double height,
               double width, string style): base(creator, name, year, height,
               width)
{
    this.style = style;
}
```

הפעולה מקבלת את הפרמטרים הנדרשים לאתחול מופע עצם מסוג ציור Painting תוך שימוש בפעולה הבונה של המחלקה Exhibit לאתחול תכונות העצם: שם יוצר, שם מוצג, שנת יצירה, גובה ורוחב מוצג. הפעולה מאתחלת בעצמה את תכונת סגנון הציור style, כיוון שאינה תכונה של מוצג כללי.

ב. ציון מנגנון תכנות מונחה עצמים הבא לידי ביטוי בהגדרת הפעולות ( ) ExhibitionSpace במחלקות Exhibit ו-Statue.

בהגדרת הפעולות ( ) ExhibitionSpace במחלקות Exhibit ו-Statue בא לידי ביטוי מנגנון הדריסה. בשתי המחלקות הפעולות מוגדרות בשם זהה, אך מימושן שונה. הפעולה משמשת לחישוב שטח התצוגה הנדרש להצגת מוצג בתערוכה, אך אופן החישוב תלוי ומשתנה בהתאם לסוג המוצג. במחלקה מוצג (כללי) Exhibit החישוב מתבסס על תכונות הגובה והרוחב, אך במחלקה פסל Statue החישוב מתבסס גם על תכונת העומק. המנגנון המאפשר, בתכנות מונחה עצמים, להגדיר פעולות בשם זהה אך עם מימוש שונה במחלקות עם יחס של ירושה ביניהן, הוא מנגנון הדריסה (override). מימוש הפעולה במחלקת הבסיס מוצג (כללי) Exhibit נדרס ע"י מימוש הפעולה במחלקה הנגזרת פסל Statue. במחלקת הבסיס הפעולה מוגדרת ע"י מילת המפתח virtual, ובמחלקה הנגזרת - ע"י מילת המפתח override.

מימוש הפעולה במחלקה Exhibit:

```
public virtual double ExhibitionSpace()
{
    return this.height * this.width;
}
```

מימוש הפעולה במחלקה Statue :

```
public override double ExhibitionSpace ()
{
    return this.height * this.width * this.depth;
}
```

ג. האם ניתן לממש את הפעולה ( ) ExhibitionSpace במחלקה Statue בעזרת הפעולה ( ) ExhibitionSpace שבמחלקה Exhibit ?

כן ניתן. במימוש הפעולה ( ) ExhibitionSpace במחלקה Statue ניתן לקרוא לפעולה ( ) ExhibitionSpace ממחלקת הבסיס Exhibit כדי לחשב את מכפלת הגובה ברוחב באופן הבא :

```
public override double ExhibitionSpace ()
{
    return base.ExhibitionSpace () * this.depth;
}
```

לשם השוואה, ניתן לראות מעלה את אופן מימוש הפעולה לפני השינוי (מסומן בהדגש צהוב).

ד. מגדירים את המחלקה חדר Room, המייצגת חדר במוזיאון, שבו עד 25 מוצגים משלושה סוגים אפשריים: ציור, פסל וצילום. נדרש לכתוב מימוש למחלקה חדר, הכולל את כותרת המחלקה, הגדרת התכונות, מימוש פעולה המקבלת מוצג ומוסיפה אותו לחדר, ומימוש פעולה המחזירה את מספר הציורים שנמצאים בחדר :

```
class Room
{
    private Exhibit[] exhibits = new Exhibit[25];

    private int numOfExhibitsInRoom = 0;

    public void AddExhibitToRoom(Exhibit exhibit)
    {
        exhibits[this.numOfExhibitsInRoom++] = exhibit;
    }

    public int NumOfPaintingsInRoom ()
    {
        int numOfPaintingsInRoom = 0;

        for (int i = 0; i < this.numOfExhibitsInRoom; i++)
            if (this.exhibits[i] is Painting)
                numOfPaintingsInRoom++;

        return numOfPaintingsInRoom;
    }
}
```



תכונות המחלקה הן מערך exhibits בגודל של עד 25 מוצגים ומאפיין numOfExhibitsInRoom, המייצג את מספר המוצגים שנמצאים בפועל בחדר ומאותחל ל-0. הפעולה להוספת מוצג חדש לחדר AddExhibitToRoom, מוסיפה אותו במקום הפנוי הבא במערך, בהנחה שיש מקום פנוי בחדר והפעולה לאחזור מספר הציורים בחדר NumOfPaintingsInRoom מבצעת מעבר כולל על המוצגים שנמצאים בפועל בחדר, ומאתרת את הציורים מביניהם בעזרת האופרטור is.

ה. מגדירים את המחלקה תערוכה Exhibition, המייצגת תערוכת אומנות המתקיימת במוזיאון ב-10 חדרים הממוספרים מ-1 עד 10. נדרש לכתוב מימוש למחלקה תערוכה, הכולל את כותרת המחלקה, הגדרת התכונות, מימוש פעולה המקבלת מספר חדר בין 1 ל-10 ומוצג ומוסיפה אותו לחדר המתאים ומימוש פעולה המקבלת מספר חדר בין 1 ל-10 ומחזירה את מספר הציורים בחדר:

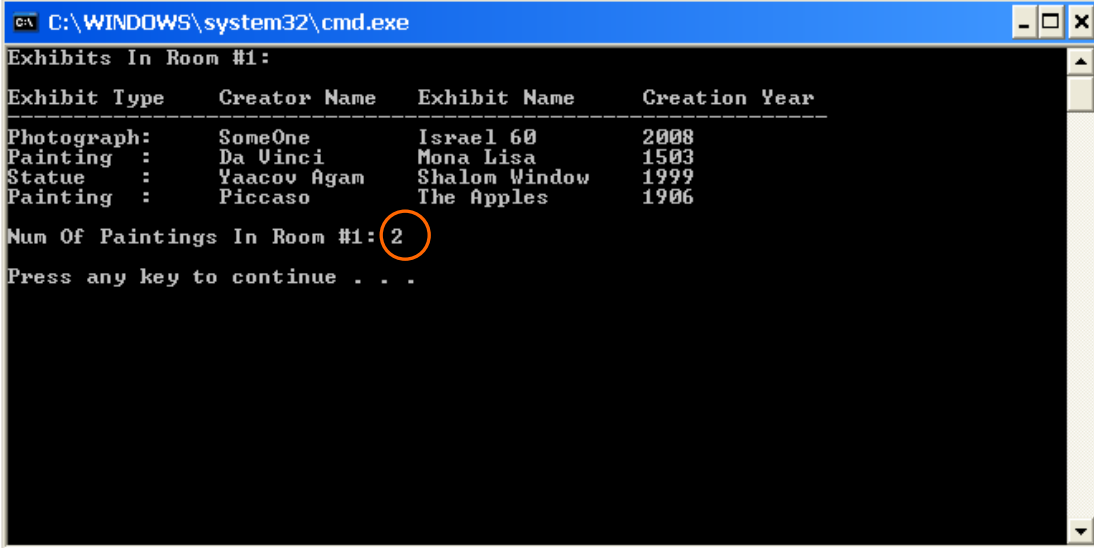
```
class Exhibition
{
    private Room[] rooms = new Room[10];

    public void AddExhibitToRoom(int numOfRoom, Exhibit exhibit)
    {
        this.rooms[numOfRoom - 1].AddExhibitToRoom(exhibit);
    }

    public int NumOfPaintingsInRoom(int numOfRoom)
    {
        return this.rooms[numOfRoom - 1].NumOfPaintingsInRoom();
    }
}
```

ניתן לראות בהגדרת המחלקה שתכונת המחלקה היא rooms - מערך של 10 חדרים, ושהפעולות להוספת מוצג לחדר בתערוכה ואחזור מספר הציורים בחדר בתערוכה משתמשות בפעולות המתאימות של המחלקה חדר Room בהנחות שפרמטר מספר החדר הניתן לפעולות אלו צריך להכיל מספר בין 1 ל-10, ושהחדר אליו נדרש להוסיף מוצג חדש – פנוי.

להדגמת תוכנית, המתבססת על היררכית מחלקות המוצגים ועל המחלקות המייצגות חדר ותערוכה במוזיאון, נאתחל מופע עצם מסוג תערוכה Exhibition, ונוסיף לחדר מספר 1 בתערוכה 4 מוצגים מסוגים שונים. עבור המוצגים הנמצאים בחדר זה נקבל את הפלט:



```
C:\WINDOWS\system32\cmd.exe
Exhibits In Room #1:
Exhibit Type      Creator Name      Exhibit Name      Creation Year
-----
Photograph:      SomeOne          Israel 60         2008
Painting :       Da Vinci         Mona Lisa         1503
Statue :         Yaacov Agam     Shalom Window    1999
Painting :       Picasso          The Apples        1906
Num Of Paintings In Room #1: 2
Press any key to continue . . .
```

הפלט מציג את פרטי 4 המוצגים בחדר מספר 1 בתערוכה: סוג מוצג, שם יוצר, שם מוצג ושנת יצירה. ניתן לראות שמספר הציורים בחדר הוא 2 – ה"מונה ליזה" של דה וינצ'י ו"התפוחים" של פיקאסו.

**23.** שאלה העוסקת בפיתוח מחלקות לייצוג טלפונים קוויים וניידים בפרויקט של חברת טלפונים.

א. הוספת מחלקה חדשה לפרויקט, שתטפל בתכונות ופעולות, המשותפות לטלפונים קוויים וניידים. יש לכתוב את כותרת המחלקה ולכתוב במילים את הפעולות שיש לכלול בה ואת התכונות שיש להגדיר בה כך שיהיה אפשר לממש פעולות אלה.

המחלקה החדשה תיקרא BasePhone ותייצג טלפונים קוויים וניידים. המחלקה תהווה מחלקת בסיס למחלקה המייצגת טלפון קווי Phone ולמחלקה המייצגת טלפון נייד CellPhone. שתי מחלקות אלו תהיינה המחלקות הנגזרות של המחלקה החדשה.

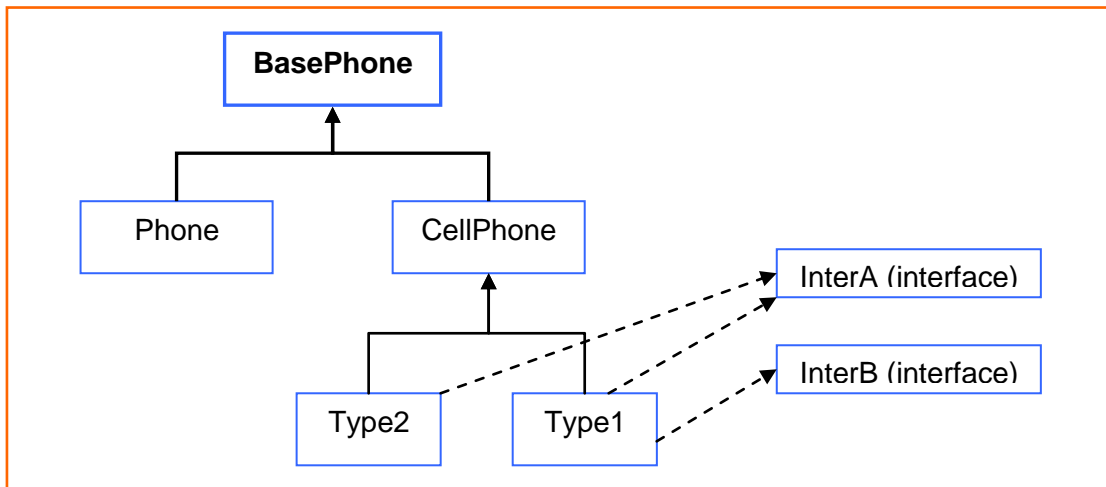
מחלקת הבסיס החדשה תהיה אבסטרקטית, כיוון שהיא מגדירה טיפוס אב כללי לטיפוסי הטלפון הקווי והנייד. המשמעות היא שלא ניתן ליצור אובייקטים ממחלקה זו, אלא רק מהמחלקות הישירות המייצגות טלפון קווי ואת שני סוגי הטלפון הנייד. לפיכך, כותרת המחלקה תראה כך:

```
abstract class BasePhone
```

מהגדרת הפעולות שאפשר לבצע על כל אחד מסוגי הטלפון ניתן לראות שפעולות החיוג וקבלת השיחה הינן משותפות לכל סוג של טלפון - קווי או נייד. לפיכך שתי פעולות אלה תופענה במחלקת הבסיס החדשה. כדי לאפשר מימוש בפועל של שתי פעולות אלה נדרש להגדיר שתי תכונות: מספר הטלפון ומצב הטלפון. תכונת מספר הטלפון קשורה כמובן לטלפונים בעולם האמיתי, אך גם בשאלה זו היא נדרשת במימוש פעולות החיוג וקבלת השיחה, כדי לאפשר, מצד אחד, את ציון יעד השיחה עבור מחייג השיחה, ומצד שני, את זיהוי מקור השיחה אצל מקבל השיחה. תכונת מצב הטלפון נדרשת כדי לשנות ולהתאים את מצב הטלפון לפעולה - חיוג או קבלת שיחה. מעצם קיום פעולות החיוג וקבלת השיחה ניתן ללמוד שהטלפון יכול להימצא בלפחות 2 מצבים שונים: מחייג ומקבל שיחה. כתוצאה מביצוע פעולות החיוג וקבלת שיחה הטלפון עובר למצב תפוס. כאשר הטלפון אינו מחייג או מקבל שיחה ואינו תפוס, כתוצאה מפעולות אלו, אז הטלפון במצב זמין. להלן תרשים המציין את תכונות ופעולות המחלקה:

<code>abstract class BasePhone</code>
מספר הטלפון [מחרוזת]
מצב הטלפון [רשימת ערכים: זמין, מחייג, מקבל שיחה, תפוס]
חיוג
קבלת שיחה

ב. הוספת המחלקה החדשה מסעיף א' לתרשים המחלקות ; יש לצייר את הקשרים החדשים שנוספים ולהסביר אותם במילים.



המחלקה BasePhone הוספה לתרשים כמחלקת אב למחלקות Phone ו-CellPhone המייצגות טלפון קווי וטלפון נייד בהתאמה. הקשרים שנוספו מציינים שהמחלקות Phone ו-CellPhone יורשות מהמחלקה החדשה BasePhone [המחלקה החדשה והקשרים החדשים מופיעים בצורה מובלטת בתרשים].

ג. עבור כל מחלקה נגזרת בתרשים המחלקות יש לרשום כותרת ולכתוב במילים את הפעולות שיש לכלול בה ואת התכונות שיש להגדיר בה כך שיהיה אפשר לממש פעולות אלה.

### המחלקה Phone

המחלקה, המייצגת טלפון קווי, יורשת ממחלקת הבסיס BasePhone, לפיכך בכותרתה יצוין קשר הירושה בין המחלקה הנגזרת למחלקת הבסיס שלה:

```
class Phone : BasePhone
```

פעולות החיוג וקבלת השיחה, המאפיינות טלפון קווי, כבר מוגדרות במחלקת הבסיס BasePhone, לפיכך לא נדרש להגדירן במחלקה המייצגת טלפון קווי. למעט שתי פעולות אלו לא נדרש, ע"פ הגדרת השאלה, להגדיר פעולות נוספות המאפיינות טלפון קווי. כך גם, אין צורך להגדיר תכונות נוספות מעבר לתכונות מספר הטלפון ומצב הטלפון שהוגדרו במחלקת הבסיס.

## המחלקה CellPhone

המחלקה, המייצגת טלפון נייד, יורשת גם היא ממחלקת הבסיס BasePhone, ומהווה מחלקת בסיס למחלקות, המייצגות את סוגי הטלפון הנייד Type1 ו-Type2. לפיכך, מחלקה זו תהיה אבסטרקטית, כיוון שמגדירה טיפוס אב כללי לשני טיפוסים הטלפון הנייד. המשמעות היא שלא ניתן ליצור אובייקטים ממחלקה זו, אלא רק מהמחלקות הישירות המייצגות את טיפוסים הטלפון הנייד. לפיכך, כותרת המחלקה תראה כך:

```
abstract class CellPhone : BasePhone
```

מהגדרת הפעולות הקשורות לטלפונים הניידים, ניתן לראות שפעולות לניהול זיכרון מספרי הטלפון ופעולות הקשורות בשעון הינן משותפות לשני סוגי הטלפון הנייד Type1 ו-Type2. לפיכך, הפעולות לניהול זיכרון מספרי הטלפון – הוספת מספר ומחיקת מספר - תוגדרנה במחלקת הבסיס המייצגת טלפון נייד CellPhone. כדי לאפשר מימוש בפועל של פעולות הוספה ומחיקה של מספרי טלפון לזיכרון / מהזיכרון נדרש להגדיר תכונה בשם אלפון [ספר טלפונים], שתיוצג ע"י מבנה נתונים לאחסון זיכרון מספרי הטלפון.

הפעולות הקשורות בשעון לא תוגדרנה במחלקת הבסיס לטלפונים הניידים, אלא במחלקות הנגזרות של סוגי הטלפון הנייד Type1, Type2, כיוון ששתיהן, כל אחת בנפרד, יורשות ישירות מממשק InterA, המגדיר את פעולות השעון. יחד עם זאת, היות שלא ניתן להגדיר תכונות / שדות בממשק (interface), רכיב השעון עצמו יוגדר במחלקת הבסיס, המייצגת טלפון נייד CellPhone, כיוון שהינו משותף לשני סוגי הטלפון הנייד. רכיב השעון ייוצג ע"י מבנה struct, הכולל שלושה שדות נומריים לייצוג שלושת מרכיבי השעה: שעות, דקות, שניות. להלן תרשים המציג את תכונות ופעולות המחלקה המייצגת טלפון נייד:

<code>abstract class CellPhone : BasePhone</code>
אלפון / ספר טלפונים [מערך רשימה של מספרי טלפון] רכיב שעון [מבנה עם שלושה שדות מסוג שלם: שעות, דקות, שניות]
הוספת מספר טלפון לאלפון הטלפונים מחיקת מספר טלפון מאלפון הטלפונים

## המחלקה Type1

המחלקה, המייצגת טלפון נייד מסוג Type1, יורשת מהמחלקה, המייצגת טלפון נייד CellPhone ומהממשקים, המגדירים את הפעולות הקשורות בשעון ובתמונה: InterA ו-InterB בהתאמה. לפיכך, כותרת המחלקה תראה כך:

```
class Type1 : CellPhone, InterA, InterB
```

המחלקה מממשת את פעולות הממשקים: הצגת שעות וכיוון שעות של ממשק InterA וצילום תמונה והצגת תמונה של ממשק InterB. הפעולות הקשורות בשעות פועלות על רכיב השעות המוגדר במחלקת הבסיס המייצגת טלפון נייד, לפיכך לא תוגדרנה במחלקה זו תכונות. להלן תרשים המציג את פעולות המחלקה:

<pre>class Type1 : CellPhone, InterA, InterB</pre>
- אין תכונות -
הצגת שעות
כיוון שעות
צילום תמונה
הצגת תמונה

### המחלקה Type2

המחלקה, המייצגת טלפון נייד מסוג Type2, יורשת מהמחלקה המייצגת טלפון נייד CellPhone ומהממשק המגדיר את פעולות השעות InterA. לפיכך, כותרת המחלקה תראה כך:

```
class Type2 : CellPhone, InterA
```

המחלקה מממשת את פעולות השעות - הצגת שעות וכיוון שעות - של ממשק InterA, הפועלות על רכיב השעות המוגדר במחלקת הבסיס המייצגת טלפון נייד. לפיכך, לא תוגדרנה תכונות במחלקה זו. להלן תרשים המציג את פעולות המחלקה:

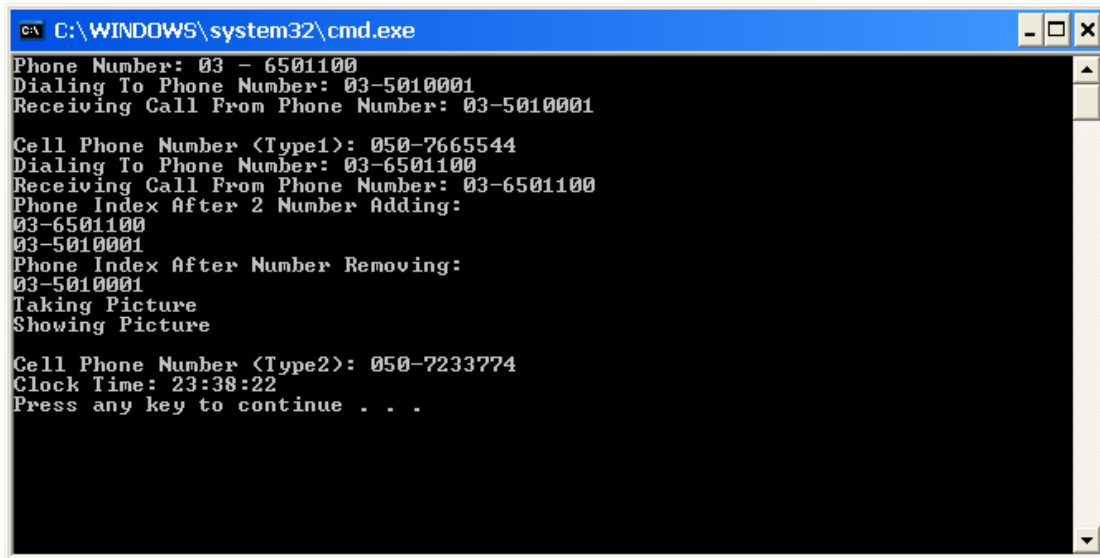
<pre>class Type2 : CellPhone, InterA</pre>
- אין תכונות -
הצגת שעות
כיוון שעות

ד. הוספת מחלקה חדשה, המייצגת נקודת שירות של החברה, שבה טלפונים מסוגים שונים אשר נמצאים בתיקון. יש לכתוב את כותרת המחלקה, את תכונותיה ותיעוד עבורן.

המחלקה החדשה תיקרא PhonesInRepair (טלפונים בתיקון) ותכיל תכונה, בשם phonesInRepairList, המייצגת רשימה של טלפונים מסוגים שונים - קווי או נייד - הנמצאים בתיקון בנקודת השירות. הרשימה תיוצג ע"י טיפוס הנתונים ArrayList, שיכיל אובייקטים מטיפוס המחלקה BasePhone, המייצגת טלפונים קוויים וניידים.

```
class PhonesInRepair
{
    //Array List With 'BasePhone' Class Elements
    private ArrayList phoneInRepairList;
}
```

להדגמת תוכנית, המבוססת על היררכיית מחלקות הטלפונים, נאתחל שלושה מופעי עצם, המייצגים טלפון קווי וטלפונים ניידים מסוג Type1 ו-Type2, נפעיל באמצעותם פעולות אפשריות לביצוע בהתאם לסוג הטלפון, ונקבל את הפלט הבא:



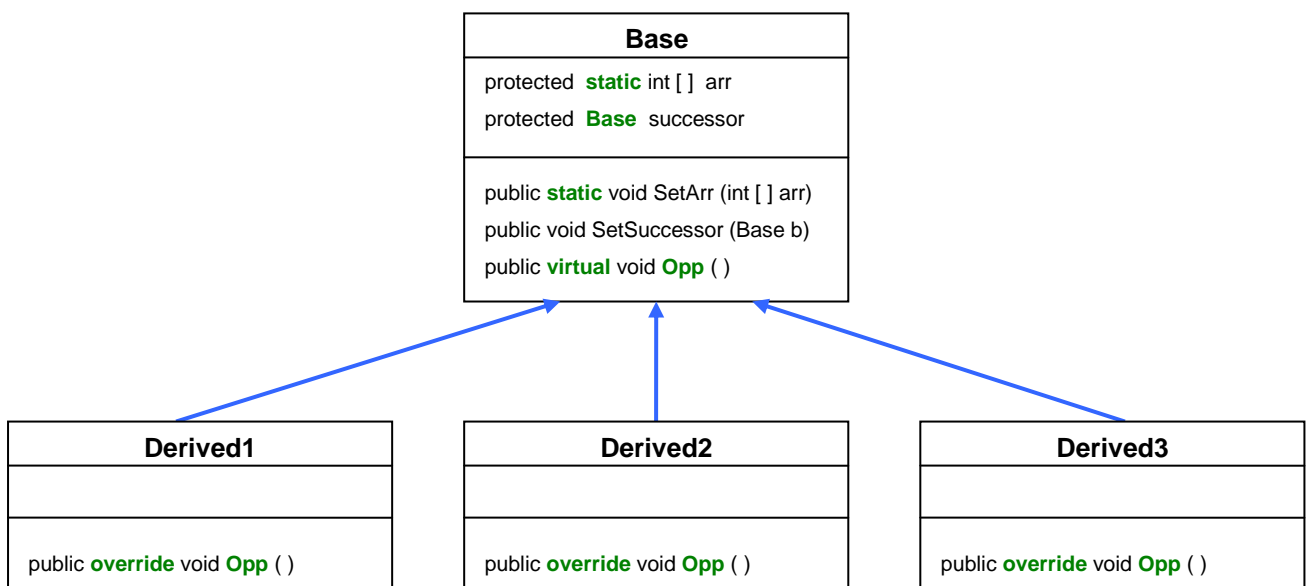
```
C:\WINDOWS\system32\cmd.exe
Phone Number: 03 - 6501100
Dialing To Phone Number: 03-5010001
Receiving Call From Phone Number: 03-5010001
Cell Phone Number <Type1>: 050-7665544
Dialing To Phone Number: 03-6501100
Receiving Call From Phone Number: 03-6501100
Phone Index After 2 Number Adding:
03-6501100
03-5010001
Phone Index After Number Removing:
03-5010001
Taking Picture
Showing Picture
Cell Phone Number <Type2>: 050-7233774
Clock Time: 23:38:22
Press any key to continue . . .
```

בפלט המתקבל ניתן לראות מספרי טלפון (פיקטיביים) של טלפון קווי ושני טלפונים ניידים מסוג Type1 ו-Type2. עבור הטלפון הקווי מוצג הפלט, המתקבל בעת ביצוע פעולות חיוג וקבלת שיחה. עבור טלפון נייד מסוג Type1 מוצג הפלט, המתקבל בעת ביצוע פעולות חיוג, קבלת שיחה, צילום תמונה והצגת תמונה, ומוצג אלפון הטלפונים לאחר הוספת 2 מספרי טלפון לזיכרון והסרת מספר טלפון מהזיכרון. עבור טלפון נייד מסוג Type2 מוצגת השעה שמראה השעון במכשיר.

24. השאלה עוסקת בפרויקט הכולל את המחלקות: Base, Derived1, Derived2, Derived3 ו-MainApp. נדרש לבצע מעקב לתוכנית המבוססת על מחלקות הפרויקט.

א. כתיבת מעקב אחר הפעולה Main במחלקה העיקרית MainApp של הפרויקט, ורישום הפלט המתקבל. מעקב הכולל את ערכי המשתנים ואת ערכי התכונות של כל עצם.

ראשית נתאר את היררכיית המחלקות בפרויקט. מלבד המחלקה העיקרית MainApp, הכוללת את פעולת התוכנית הראשית Main, כולל הפרויקט את מחלקת הבסיס Base, ממנה יורשות שלוש המחלקות הנגזרות Derived1, Derived2, Derived3, כמתואר בתרשים המחלקות הבא:



מתרשים המחלקות ניתן לראות שמחלקת הבסיס Base מכילה תכונה סטטית מסוג מערך של שלמים (arr), העוברת בירושה למחלקות הנגזרות, ומשותפת לכל מופעי העצם, הנוצרים ממחלקה זו או מכל אחת ממחלקותיה הנגזרות. נוסף לתכונת המערך, מכילה מחלקת הבסיס תכונה (successor) מטיפוס מחלקת הבסיס. תכונה שאינה סטטית, אשר עוברת בירושה למחלקות הנגזרות. המשמעות היא שתכונה זו מכילה ייחוס [הצבעה] למופע עצם, הנוצר ממחלקת הבסיס או מכל אחת ממחלקותיה. הדבר ייתכן, כיוון שמשנתנה מטיפוס מחלקת בסיס יכול להתייחס במופע עצם מטיפוס מחלקה נגזרת. מחלקת הבסיס מכילה גם מתודות לאתחול של שתי תכונות המחלקה.

מלבד עקרון הירושה, בהיררכיית המחלקות בפרויקט בא לידי ביטוי גם עיקרון הפולימורפיזם, המיושם בעזרת מנגנון הדריסה (override). במחלקת הבסיס מוגדרת מתודה וירטואלית (virtual) בשם Opp(), שמימשה נדרס ומוגדר באופן שונה בכל אחת מהמחלקות הנגזרות. המשמעות היא שכאשר מוגדר עצם מטיפוס מחלקת הבסיס, המאותחל למופע של אחת



המחלקות הנגזרות, יופעל מימוש המתודה ( ) Opp של המחלקה הנגזרת בהתאם לטיפוס מופע העצם המוקצה בפועל בזמן ריצת התוכנית.

בהתאם לניתוח היררכיית המחלקות, נערוך מעקב אחר ביצוע קטעי הקוד של המתודה Main במחלקה העיקרית בפרויקט MainApp.

### אתחול מופעי העצם והתכונות

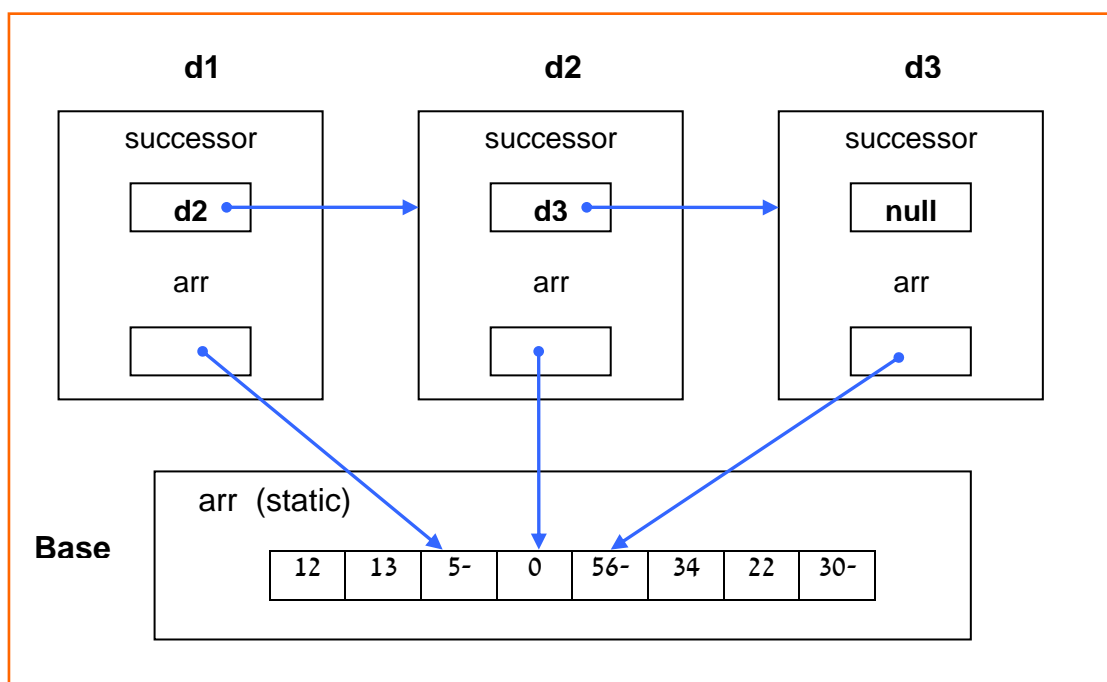
```

Derived1 d1 = new Derived1 ();
Derived2 d2 = new Derived2 ();
Derived2 d3 = new Derived2 ();

d1.SetSuccessor(d2);
d2.SetSuccessor(d3);

int[] arr = { 12, 13, -5, 0, -56, 34, 22, -30 };
Base.SetArr(arr);
    
```

בתוכנית מתבצע אתחול של מופעי העצם d1, d2, d3 מטיפוסי המחלקות הנגזרות Derived1, Derived2, Derived3 בהתאמה. תכונת successor של מופע העצם d1 (מטיפוס Derived1) מאותחלת בייחוס למופע העצם d2 (מטיפוס Derived2), ותכונת successor של מופע העצם d2 מאותחלת בייחוס למופע העצם d3 (מטיפוס Derived2), ותכונת successor של מופע העצם d3 מאותחלת בייחוס למופע העצם null. תכונת המערך הסטטית של מחלקת הבסיס Base, מאותחלת במערך של 8 מספרים שלמים. תכונה אשר משותפת ונגישה למופעי העצם d1, d2, d3. נתאר בתרשים את שלושת מופעי העצם לאחר הקצאתם ואתחול תכונותיהם.



## ביצוע המתודה ( ) Opp

ננתח את ביצוע הפעלת המתודה ( ) Opp על מופע העצם d1:

d1.Opp();

בעקבות הוראה זו תופעל גרסת המתודה של המחלקה Derived1, כיוון שמופע העצם d1 הינו מטיפוס מחלקה זו. לפיכך, נערוך מעקב אחר ביצוע המתודה ( ) Opp של המחלקה Derived1 עבור תכונות מופע העצם d1:

Base.arr.Length	i	i < Base.arr.Length	Base.arr [i]	Base.arr [i] > 0	sum	פלט	this.successor
					0		
8	0	true	12	true	12		
	1	true	13	true	25		
	2	true	-5	false			
	3	true	0	false			
	4	true	-56	false			
	5	true	34	true	59		
	6	true	22	true	81		
	7	true	-30	false			
	8	false					
						Sum: 81	
							d2

```
if (this.successor != null)
    this.successor.Opp();
```

מתרשים המופעים ומטבלת המעקב ניתן לראות שערך תכונת successor של מופע העצם d1 הוא d2 ואינו null, לפיכך ביצוע בלוק התנאי שבסוף מימוש המתודה ( ) Opp של המחלקה Derived1, יוביל להפעלת המתודה ( ) Opp על מופע העצם המאותחל d2. במקרה זה תופעל גרסה המתודה של המחלקה Derived2, כיוון שמופע העצם d2 הינו מטיפוס מחלקה זו. לפיכך, נערוך מעקב אחר ביצוע המתודה ( ) Opp של המחלקה Derived2 עבור תכונות מופע העצם d2:

this.successor	פלט	counter	Base.arr [i] <= 0	Base.arr [i]	i < Base.arr.Length	i	Base.arr.Length
		0					
			false	12	true	0	8
			false	13	true	1	
		1	true	-5	true	2	
		2	true	0	true	3	
		3	true	-56	true	4	
			false	34	true	5	
			false	22	true	6	
		4	true	-30	true	7	
					false	8	
	<b>Counter: 4</b>						
<b>d3</b>							

```
if (this.successor != null)
    this.successor.Opp();
```

מתרשים המופעים ומטבלת המעקב ניתן לראות שערך תכונת successor של מופע העצם d2 הוא d3 ולא null. לפיכך, התנאי בסוף מימוש המתודה ( ) Opp של המחלקה Derived2 מתקיים, וגרסת המתודה ( ) Opp של המחלקה Derived3 תופעל על מופע העצם המאותחל d3. נערוך מעקב אחר ביצוע המתודה עבור תכונות מופע העצם d3:

this.successor	פלט	counter	Base.arr [i] == 0	Base.arr [i] % 2 == 0	Base.arr [i]	i < Base.arr.Length	i	Base.arr.Length
		0						
	12	1	false	true	12	true	0	8
	13		false	false	13	true	1	
	-5		false	false	-5	true	2	
	*	2	true	true	0	true	3	
	-56	3	false	true	-56	true	4	
	34	4	false	true	34	true	5	
	22	5	false	true	22	true	6	
	-30	6	false	true	-30	true	7	
						false	8	
	<b>Counter: 6</b>							
<b>null</b>								

מתרשים המופעים ומטבלת המעקב ניתן לראות שערך תכונת successor של מופע העצם d3 הוא null, ולכן הוראת הביצוע `this.successor.Opp()` - בבלוק התנאי שבסוף מימוש המתודה `Opp()` של המחלקה `Derived3` - לא תבצע. כך למעשה, מסתיים ביצוע ההוראה האחרונה `d1.Opp()` שבמתודה העיקרית `Main` ושתי הפעולות המשנה שבאו בעקבותיה.

ניתן להשוות את הפלטים המופיעים בעמודות הפלט בטבלאות המעקב לפלט מרוכז המתקבל מהרצת התוכנית בסביבת הפיתוח:

A screenshot of a Windows command prompt window. The title bar reads "C:\WINDOWS\system32\cmd.exe". The window content shows the following text:

```
Sum:81
Counter: 4
12
13
-5
*
-56
34
22
-30
Counter: 6
Press any key to continue . . .
```

ב. משנים את מימוש הפעולה `Main` במחלקה `MainApp`; נדרש לקבוע אם הטענה לפיה - ביצוע הפעולה `Main` לאחר שינוי המימוש אינו מסתיים - נכונה.

ננתח את קטעי הקוד של הפעולה `Main` לאחר השינוי במימוש:

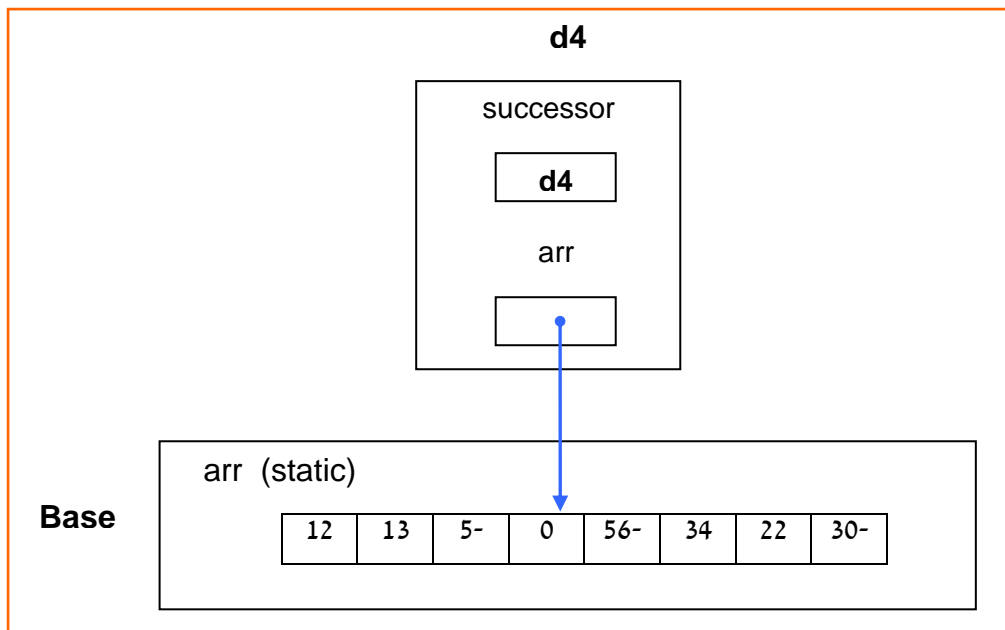
## אתחול מופעים

```
Derived1 d4 = new Derived1();
d4.SetSuccessor(d4);

int[] arr = { 12, 13, -5, 0, -56, 34, 22, -30 };
Base.SetArr(arr);
```

בתוכנית מתבצע אתחול של מופע עצם `d4` מטיפוס המחלקה הנגזרת `Derived1`. תכונת `successor` של מופע העצם מאותחלת בייחוס למופע העצם עצמו, `d4`. תכונת המערך הסטטית של מחלקת הבסיס `Base`, מאותחלת במערך של 8 מספרים שלמים בדומה למימוש הפעולה בסעיף א.

נתאר בתרשים את מופע העצם לאחר הקצאתו ואתחול תכונותיו :



### ביצוע המתודה ( ) Opp

ננתח את ביצוע הפעלת המתודה ( ) Opp על מופע העצם d4 :

```
d4.Opp();
```

בעקבות הוראה זו תופעל גרסת המתודה של המחלקה Derived1, כיוון שמופע העצם d4 הינו מטיפוס מחלקה זו. בסוף מימוש המתודה מופיע בלוק תנאי הבדוק אם תכונת successor של מופע העצם אינה null :

```
if (this.successor != null)
    this.successor.Opp();
```

מניתוח קטע האתחול של מופע העצם ומתרשים המופע ניתן לראות שערך תכונת successor של מופע העצם d4 אינו null אלא מופע העצם עצמו d4. לפיכך, התנאי לעולם מתקיים ותחל סדרת הפעלות אינסופית של המתודה ( ) Opp על מופע העצם d4 [בכל הפעלה של המתודה ערך תכונת successor של מופע העצם d4 נותר d4 ואינו null]. הדבר יגרום לכך שביצוע הפעולה Main לא יסתיים לעולם ולכן הטענה נכונה.