**מבנה נתונים ותכנות מונחה עצמים**

מבחן מגן 2 תשפ"ד

|  |  |
| --- | --- |
| **שם המחלקה :**  **הנדסת תכנה** |  |
| **כיתה: ג9**  **קבוצה א +קבוצה ב** | **שם המרצה:**  ***יזהר משקיף***  ***שפיק סייד-אחמד***  ***רחל שליסלברג*** |
| **תאריך המבחן:**  **9.4.24** | **שעת תחילת בחינה:**  **9.00** |
| **משך הבחינה:**  **4 שעות** | **האם יש צורך במחברת:  כן, 2 מחברות לכל סטודנט** |
| **חומר עזר מותר:**   1. **מחשבון (אין להשתמש במחשב כף יד או מחשבון עם תקשורת. חיצונית)** 2. **קלסר אחד בלבד עם חומר הרצאות. אין להוציא דפים מהקלסר.** |  |
| **הוראות לנבחן :**  **שאלות 14: יש לענות על 3 שאלות בלבד – כל שאלה 16 נקודות. חלק ב – 36 נקודות שאלות 5-8: יש לענות על 2 שאלות בלבד – כל שאלה 18 נקודות. חלק ג – 16 נקודות שאלות 10-9: יש לענות על 1 שאלות בלבד – כל שאלה 16 נקודות.**  **נקודה אחת תינתן על הערכה.**  **סה"כ:100 נקודות** |  |

* **שמור על טוהר הבחינות ועל הנחיות הבוחנים והקפד שטלפון הנייד נמצא בתיקך.**
* **הקפד לרשום על גבי מחברת המבחן את שמך המלא בכתב קריא וברור.**
* **רשום תשובותיך בעט כחול או שחור בלבד .**
* **השימוש בנוזל מחיקה אסור.**
* **אין לגלוש מעבר לשוליים המסומנים במחברת המבחן.**
* **החל פתרון לכל שאלה בדף חדש וציין בראשו את מס' השאלה.**
* **העבר קו אלכסוני למחיקה של דפים או חלקי תשובה שאינך רוצה שיבדקו.**
* **אין להעביר חומר עזר בין הנבחנים.**

חשוב מאוד:

**יש לענות במחברות נפרדות על השאלות של מבנה נתונים ושל תכנות מוחה עצמים.**

**רשום על כל מחברת מבנה נתונים + שם המרצה / תכנות מונחה עצמים**

**בהצלחה !!**

**חלק א' – בחר 3 מתוך 4 שאלות (1-4)**

**שאלה 1:** (16 נקודות) – ***מבנה נתונים: רשימה מקושרת***

"שרשרת מיוחדת" היא רשימה מקושרת של מספרים שלמים, הכוללת שני חלקים משורשרים.

**חלקה הראשון:** **ממויין** מקטן לגדול ומכיל מספר **אי-זוגי** של איברים.

**חלקה השני מכיל 3 איברים**, והם :

1. **סכום** החלק הראשון של הרשימה,

2. **מספר** האיברים count שבחלק הראשון,

3. ערך האיבר **במקום האמצעי**.

כתוב פעולה שמקבלת הפנייה לחוליה הראשונה ברשימה המקושרת רגילה ויוצרת "שרשרת מיוחדת" לפי התנאים הבאים:

אם אורך הרשימה זוגי או הרשימה אינה ממויינת, אזי הפעולה לא משנה את הרשימה ומדפיסה הודעה מתאימה,

אחרת (אורכה אי-זוגי וממוינת) **מוסיפה 3 איברים** הנדרשים כדי להפוך אותה לרשימה מיוחדת.

כותרת הפעולה:

public static Node<int> **MakeSpecial**(Node<int> lst)

**דוגמה:**

נתונה הרשימה:

lst: ->10->30 ->30->45->50->51->60-||

מוחזרת הרשימה:

lst: ->10->30 ->30->45->50->51->60**->276->7->45-||**

כאשר **276** הוא סכום איברי הרשימה, **7** הוא אורך הרשימה, והאיבר אמצעי הוא **45**.

**שאלה 2:** (16 נקודות) – ***מבנה נתונים: רשימה מקושרת דו כיוונית ומחסנית***

1. (12 נק')

כתוב פעולה חיצונית שמקבלת **רשימה מקושרת דו כיוונית** של מספרים שלמים (שרשרת שלמים) ומחזירה מחסנית שמכילה את כל האיברים המופעים ברשימה יותר מפעם אחת

שימו לב הרשימה אינה משתנה ובמחסנית כל איבר יופיע פעם אחת בלבד.

static Stack<int> **Collect**(BinNode<int> lst)

**דוגמה**: נתונה הרשימה

lst 🡪**22=21=14=13=35=14=14=14=11=36=35=75=46=11-=**

תוחזר המחסנית:

stk: (top) 🡪11, 35, 14**]**

1. (4 נק') מה הסיבוכיות של הפעולה שכתבת?

**שאלה 3:** (16 נקודות) – ***מונחה עצמים:***

נתונות ארבע המחלקות הבאות:

public class A {

protected int x;

public A() {

x = 0;

}

public virtual void Method(int num) {

x += num;

}

public override string ToString() {

return "" + x;

}

}// end of class A

public class B : A{

private int y;

public B()

{

y = 1;

}

public override void Method(int num) {

x -= num;

y += num;

}

public override string ToString () {

return "" + x + "." + y;

}

}// end of class B

public class C : A{

private int y;

public C()

{

y = 10;

}

public override void Method(int num) {

x += num;

y += num;

}

public override string ToString() {

return "" + x + "/"+ y;

}

} // end of class C

public class Test

{

public static void Main (string [] args)

{

A [] array = new A[4];

***שים לב***:

המשך השאלה בעמוד הבא

array[0] = new C();

array[1] = new B();

array[2] = new A();

array[3] = new B();

for (int i = 0; i < array.length; i ++)

{

Console.WriteLine("array[" + i + "] = " + array[i]);

array[i].Method(i+2);

Console.WriteLine("array[" + i + "] = " + array[i]);

}

}

}

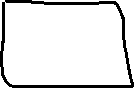
כתבו את הפלט של התוכנה. כשנריץ את פעולת Main של Test .

**בנפרד** כתבו את המעקב של כל המשתנים שבתוכה. אין צורך לציין כנגד איזה לופ זה, פשוט לציין מה קורה בכל השתנות. ציירו איור של אובייקט כולל הטיפוס, הערך שניתן לטיפוס, וערך כל אחד מהמשתנים שלו.

**לדוגמא:**

**Type: A,**

**Value: B**



**y = 5, x = 6**

**שאלה 4:** (16 נקודות) – ***מונחה עצמים***

לפניך המחלקות הבאות:

public class Vehicle//=כלי רכב

{

private int serialNum = 0;

protected int maxSpeed = 0;

public Vehicle(int serialNum)

{

this.serialNum = serialNum;

this.maxSpeed = 40;

}

public virtual void Move()

{

Console.WriteLine("Some vehicle is moving");

}

}//end

public class Rail: Vehicle

{

public Rail(int serialNum): base(serialNum)

{

this.maxSpeed = 300;

}

public override void Move()

{

Console.WriteLine("Rail is moving");

}

public string RailDoorsOpen()

{

return "Rail is openning doors";

}

}//end

public class Bus: Vehicle

{

public Bus(int serialNum) : base(serialNum)

{

this.maxSpeed = 120;

}

public override void Move()

{

Console.WriteLine("Bus is moving");

}

public string BusDoorsOpen()

{

***שים לב***:

המשך השאלה בעמוד הבא

return "Bus is openning doors";

}

}//end

א. האם בין המחלקות יש דריסת פעולות (overriding) או העמסת פעולות (overloading) . הסבירו את תשובתכם.

ב. נתונה פעולה סטטית באחת המחלקות. רישמו מה יקרה אם נריץ אותה ונחליף את המקום שכתוב \*\*\*\* באחד מקטעי הקוד הנתונים להבא.

public static void Demo()

{

Vehicle v1 = new Rail(10);

Vehicle v2 = new Bus(11);

Vehicle v3 = new Bus(12);

Vehicle v4 = new Vehicle(23);

\*\*\*\*

}

**במידה ויש פלט יש לרשום אותו.**

**יש לרשום אם זה קוד תקין, שגיאת קומפליציה או שגיאת הרצה.**

1-

v1.Move();

v4.Move();

2-

Console.WriteLine(v2.BusDoorsOpen());

3-

Console.WriteLine(((Rail)v2).RailDoorsOpen());

4-

Console.WriteLine(((Bus)v2).BusDoorsOpen());

5-

Console.WriteLine(v3 is Bus);

**חלק ב' – בחר 2 מתוך 4 שאלות (5-8)**

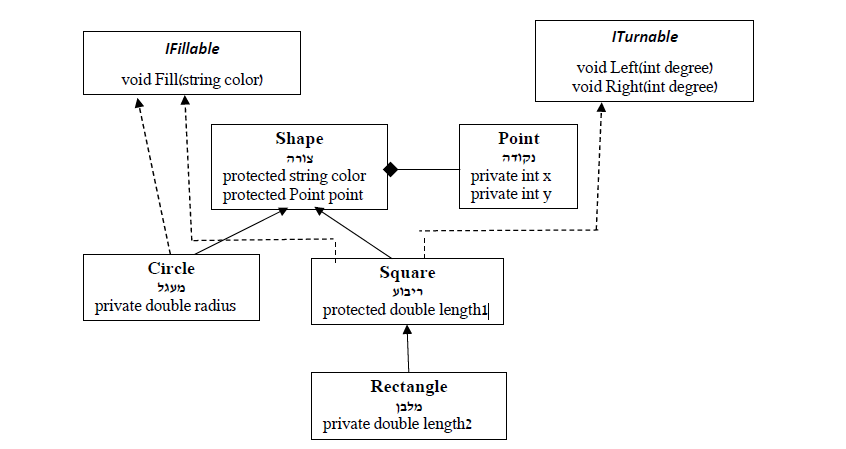
**שאלה 5:** (18 נקודות) – ***מונחה עצמים:***

נתונה היררכיית המחלקות הבאה הכוללת חמש מחלקות:

**Rectangle ,Square ,Circle ,Shape ,Point**

ושני ממשקים:

**ITurnable ו- IFillable .**



א. כתבו עבור כל מחלקה, את כותרת המחלקה ואת הכותרות של כל הפעולות ש**חייבות** להיות במחלקה.

נתון קטע קוד בפעולה הראשית (Main):

Shape s = new Square(new Point(50,50),10);

Shape c = new Circle(new Point(50,50),8);

ITurnable t;

IFillable f;

\*\*\*

ב. כתבו כותרות לכל הבנאים הנדרשים להרצה תקינה של קטע הקוד הנתון.

***שים לב***:

המשך השאלה בעמוד הבא

ג. בהתייחס לכל אח ד מהקטעים הבאים, כתבו מה יקרה בעקבות הוספתו לשיטה Main שלעיל,

במקום שמסומן בכוכביות \*\*\* .

אם הקוד אינו תקין יש להסביר ולציין את סוג השגיאה (קומפילציה או זמן ריצה(

(1) f =(IFillable) c;

(2) t = s;

t.Fill("Green");

(3) f = c;

f.Fill("Red");

(4) ((Circle)s).Left (45);

(5) Shape d = c;

d.Fill("Yellow");

(6) f = new Rectangle(new Point(10,10),10,20);

f.Left(Math.PI);

(7) c = s;

c.Left (90);

**שאלה 6:** (18 נקודות) – ***מונחה עצמים:***

פותחה תוכנה לצורך מעקב אחרי נתוני ליגת הכדורגל.

התוכנה תכיל את המחלקות הבאות:

**קבוצה (team )** שתכלול את הערכים הבאים:

- שם

- עיר

- ליגה נוכחית

- סה"כ משחקים

- משחקים ששוחקו

- נצחונות

- הפסדים

- תיקו

- נקודות

- שערים לזכות (goalsFor )

- שערים בחובה (שערים שהקבוצה ספגה) ( goalAgainst )

- הפרש שערים (goalDifferential )

**משחק ( Game) :**

**-** קבוצה א'

- קבוצה ב'

- קבוצה א' מס' שערים

- קבוצה ב' מס' שערים

**-** דקה נוכחית

- האם המשחק פעיל

**מחזור:**

***שים לב***:

המשך השאלה בעמוד הבא

**-** 6 משחקים בין כל קבוצות הליגה.

**עונה ( Season ) :**

- שנה

- הליגה (לאומית, ארצית, על, א' וכיו"ב)

- סה"כ מחזורים (rounds amount)

- המחזור הבא

- קבוצות – בכל עונה יש 12 קבוצות

- האם פעילה?

**\* אלא אם צויין אחרת – אין צורך ליצור Getters & Setters ההנחה היא שהם קיימים. כמו"כ אין צורך ליצור בנאים, אא"כ התבקש.**

**\* יש לפעול בצורה נכונה והגיונית לפי תיכנות מונחה עצמים**

**\* כללי הניקוד שישמשו אותנו הם: 3 נקודות על ניצחון, נקודה אחת על תיקו. אין נקודות על הפסד.**

א. צור את המחלקות הנ"ל עם הכותרות ועם התכונות שלהן. תן טיפוסים ושמות הגיוניים לפי מהות הדבר. תעד מעל התכונה למה היא שייכת. (בהערת קוד).

ב. צור תרשים (אין צורך בUML מלא רק בשמות המחלקות **וחצים/קווים תיקניים**) שמתאר את הקשרים בין המחלקות.

ג. צור פעולה ScoreGoal בתוך מחלקה Game שלא מחזירה כלום. היא מייצגת הבקעת גול, ע"י הקבוצה שמוגשת לה בתור פאראמטר. כתוב בתוך המתודה את כל הדברים שצריכים להתבצע בעקבות זאת (ההנחה היא שהדקות עולות מעצמן ואין צורך לטפל בהן). התחשב גם בפעולה שיש בסעיף הבא. ההנחה היא שבכל משחק יש סיום משחק, ולא ייתכן מצב שיהיה משחק מבלי שיהיה סיום משחק.

ד. צור פעולה FinishGame בתוך מחלקת Game שמייצגת סיום המשחק. גם כאן היא לא מחזירה כלום. כתוב בתוכה את כל הפעולות שאמורות להתבצע בעקבות סיום המשחק.

ה. צור פעולה סטטית שמקבלת 2 קבוצות ומחזירה את הקבוצה שמדורגת גבוה יותר בטבלה. כמובן שזה אמור לעבוד על 2 קבוצות מאותה ליגה.

אלה כללי הדירוג בין הקבוצות שישמשו אותנו, לפי סדר עדיפויות יורד. **במילים אחרות, אם הכלל הראשון לא חל - נלך לכלל השני. אם הוא כן חל – זה מספיק.**

1. הקבוצה שיש לה את מירב הנקודות
2. הקבוצה שיש לה את הפרש השערים הגבוה ביותר
3. הקבוצה שכבשה יותר שערים.
4. במידה וזה עדיין שווה – החזר את הראשונה שהוגשה.
5. בכל מקרה אחר – החזר null .

**חובה לציין באיזו מחלקה הכי הגיוני ליצור את הפעולה הזו.**

**יש לבדוק את כל הנתונים שבדרישות השאלה.**

ו. האם נכון לדעתך ליצור את הפעולה הבאה בתוך Team שמשמשת כsetter להפרש השערים. הסבר!

public void SetGoalDifferntial(int goalDifferential){

this.goalDifferential = goalDifferential;

}

**שאלה 7:** (18 נקודות) – ***מבנה נתונים: מעקב רשימה מקושרת (חד כיוונית)***

נתונה מחלקת IntList הבאה:

public class **IntList**

{

private Node<int> **head, tail**;

// שני מצביעים: אחד לראש הרשימה ואחד לזנב (איבר אחרון)

private int **length**; // אורך הרשימה

public **IntList**()

{

head = tail = null;

length = 0;

}

public void **Add**(int a)

{

head = new Node<int>(a, head);

if (tail == null) tail = head; // first element!

this.length++;

}

public override string **ToString**()

{

string st = "{";

Node<int> h = head;

while (h != null)

{

st += h.GetValue() + ",";

h = h.GetNext();

}

return st + "}" + "\nLength: " + length;

}

public bool **What1**()

{

if (this.head == null) return false;

if ( head == tail)

{

this.head = this.tail = null;

this.length--;

return true;

}

this.head = this.head.GetNext();

Node<int> pi = this.head;

while (pi.GetNext() != this.tail)

pi = pi.GetNext();

pi.SetNext(null);

tail = pi;

this.length-=2;

return true;

}

***שים לב***:

המשך השאלה בעמוד הבא

public bool **What2**(int val)

{

if (this.head == null) return false;

Node<int> h = head.GetNext();

while (h != this.tail)

{

if (h.GetValue() == val)

return true;

h = h.GetNext();

}

return false;

}

}

נתון גם קטע קוד ממחלקה ראשית Main() :

static void Main(string[] args)

{

IntList lst = new IntList();

lst.Add(20); lst.Add(30); lst.Add(60);

lst.Add(15); lst.Add(40); lst.Add(35);

lst.Add(45); lst.Add(20); lst.Add(40);

**Console.WriteLine(lst);**

int x = 45;

while (lst.What2(x))

{

lst.What1();

**Console.WriteLine(lst);**

}

}

1. (10 נקודות)

בצע מעקב אחרי התוכנית ורשום הפלט.(המעקב יהיה על Main, אין צורך להציג מעקב על הפעולות הפנימיות)

1. (4 נקודות) הסבר מה מבצעת הפעולה What1 ?
2. (4 נקודות) הסבר מה מבצעת הפעולה What2 ?

שאלה 8: (18 נקודות) ***מבנה נתונים: מעקב רקורסיבי - עץ***

נתונה פעולה Func הבאה:

**public static BinNode<int> Func(BinNode<int> root, int n)**

**{**

**if (root is null)**

**return new BinNode<int>(n);**

**int x = root.GetValue();**

**if(n < x)**

**{**

**root.SetValue(n);**

**n = x;**

**}**

**if (root.GetLeft() == null)**

**root.SetLeft(new BinNode<int>(n));**

**else if (root.GetRight() == null)**

**root.SetRight(new BinNode<int>(n));**

**else // both leafs not null, two leafs**

**{**

**if (root.GetLeft().GetValue() < root.GetRight().GetValue())**

**Func(root.GetLeft(), n);**

**else Func(root.GetRight(), n);**

**}**

**return root;**

**}**

במחלקה ראשית (Main) נתונות הפעולות הבאות

**BinNode<int> tree = null;**

**int [] ar = [100, 35, 70, 56, 90, 25, 80];**

**for(int i =0; i< ar.length; i++)**

**tree = Func(tree, ar[i]);**

הצג העץ בכל שלב בלולאה. הראה את השינוי במצביעים או בערכים(במידה ויש וישנם).

**חלק ג' – בחר 1 מתוך 2 שאלות (9-10)**

**שאלה 9:** (16 נקודות) – ***מונחה עצמים:***

**\* שאלה זו ממשיכה על שאלה 6. גם אם דילגת על שאלה 6 ההנחה היא שפתרת שם את כל מה שצריך.**

א. רוצים להכניס למערכת יכולת לספור את מס' הנוכחים באיצטדיון בכל משחק. הבעיה היא שבכל סוג איצטדיון בעולם יש שיטה אחרת לבצע זאת.

לדוגמא:

Class EnglandStadium{

public int Count(){

int someValue = 0;

//some counting implementation goes here…

Console.WriteLine("England Counting");

return someValue;

}

}

Class SpainStadium{

public int Count(){

int someValue = 0;

//some counting implementation goes here…

Console.WriteLine("Spanish Counting");

return someValue;

}

}

אנו רוצים להכניס את התכונה והפעולה הזו למערכת:

private \_\_\_\_\_\_\_ stadium;

//constructor comes here….

public int GetCrowd(){

return this.stadium.Count();

}

***שים לב***:

המשך השאלה בעמוד הבא

**\*לא ניתן לשנות קוד בתוך הפעולות שנמצאות במחלקות שמייצגות איצטדיונים! ניתן רק בשם המחלקה להוסיף דברים לפי הצורך.**

א. באיזו מחלקה היינו מכניסים את התכונה והפעולה האלו?

ב. באיזה פיתרון היית משתמש כדי שהפעולה תהיה מותאמת לכל סוגי איצטדיון שהוא ? ניתן להוסיף דברים לפי הצורך.

ג. כתוב את הפיתרון והמימוש שלו (=כלומר איך היית כותב אותו). השלם איפה שצריך את הטיפוס שצריך להיות רשום בקוד למעלה היכן שיש \_\_\_\_\_\_\_\_\_\_. (רשום במחברת).

ד. שרטט בUML **תיקני**, רק את המחלקות הנתונות שמייצגות את האיצטדיונים ואת מה שהוספת כדי לפתור את הבעיה.

ה. כתוב פעולה בתוך Season שמקבלת מס' נקודות, ומחזירה את כל הקבוצות שיש להם אותו מס' נקודות ומעלה בתור מחסנית (Stack ) של קבוצות.

ו. כתוב פעולה שמחזירה את הקבוצה שכבשה את מס' השערים הרב ביותר במשך העונה. איפה תכתוב את הפעולה הזו?

ז. לפי כללי כתיבת הקוד נראה ש

WriteLine(string str)

היא מתודה סטטית בתוך המחלקה Console.

האם נכון, ועל סמך מה את/ה אומרת את זה?

**שאלה 10:** (16 נקודות) – ***מבנה נתונים מורכב***

במכללה החליטו להוסיף למערכת הממוחשבת אפשרות לשמור את הציונים הסופיים (הממוצע המשוקלל של כל שנות הלימודים) של כל הסטודנטים הלומדים ולמדו במכללה.

לשם כך הוגדרו המחלקות Student ו-GradeFile.

כל סטודנט מאופיין באמצעות מספר סטודנט (studentId) שמורכב משמונה ספרות וציון (grade).

מכיוון שמרחב מספרי הסטודנט גדול, הוחלט שהציונים ישמרו ב – 100 רשימות, כדי לאפשר חיפוש מהיר.

מאגר ציוני הסטודנטים (GradeFile) מאופיין באצעות מערך בגודל 100 של פרטי הסטונדטים.

תהליך האחסון במאגר ייעשה באופן הבא:

לוקחים את 2 הספרות האמצעיות של מספר הסטודנט. אלה יוצרות מספר K בן 0 ל 99. מוספים את הסטודנט למאגר הנמצא במקום ה K במערך.

אם מספר הסטודנט הוא 123**45**678 אזי פרטי הסטודנט יכנסו לרשימה במקום 45 במערך

**תמונה שמכילה טקסט, צילום מסך, תרשים, גופן

התיאור נוצר באופן אוטומטי**

1. (8 נקודות)

כתבו את כותרת המחלקות Student ו-GradeFile ואת התכונות שלהן.

**הערה:** יש לבחור מבנה נתונים(תור, מחסנית, שרשרת חוליות) מתאים לשמירת פרטי הסטודנטים (מספר הסטודנטים אינו מוגבל)

1. (8 נקודות)

ממשו **במחלקה** Student את הפעולה int GetCode().

הפעולה מחזירהמספר המורכב משתי הספרות האמצעיות של מספר הסטודנט (studentId). לדוגמה: עבור מספר הסטודנט 123**45**678 הפעולה תחזיר 45.