**משימת תכנות להגשה OOP**

**הוראות:**

1. לצורך ביצוע המשימה יש ברשותכם 3.5 שעות.
2. חומר עזר: אינטרנט
3. **חל איסור מוחלט לדבר ולהיעזר בחברכם לכיתה או בכל "חבר טלפוני".**
4. **חל איסור מוחלט ליצור קשר דרך האינטרנט בין חבריכם לכיתה או מול כל גורם אחר.**
5. המשימה מהווה מבחן!
6. התייחסו למשימה כחלק מראיון עבודה.
7. זכרו, לא רק התוצאה משקפת את היכולת והציון שלכם אלא גם הדרך, צורת הכתיבה, Best Practice סדר וארגון בקוד וכדומה.

במשימה זו תייצרו מערכת מלאה של ניהול קבוצת כדורסל בליגת ה NBA, בליגה יש קבוצות לכל קבוצה יש מאמן וחמישה שחקנים, מערך משחק וליגה בהם הם משחקים.

בהרצת התוכנית ייווצר אובייקט ממחלקת League ועבורו יאותחלו 4קבוצות שונות עם השחקנים ושמותיהם בסיום האתחול כל שני קבוצות רנדומליות יתחילו לשחק כ 5 משחקים כשבכל משחק כל קבוצה מקבלת 50 תורות.  
בסיום שני המשחקים הראשונים שני הקבוצות המנצחות יתחרו ביניהם על האליפות.  
בתום כל משחק יודפסו נתונים סטטיסטים על אותו משחק ובתום 5 המשחקים יודפסו סטטיסטיקות שונות שיפורטו בהמשך.

המטרה של משימה זו היא להדפיס נתונים סטטיסטיים על הליגה ועל המשחקים של הקבוצות בה כדוגמת כמות ניצחונות, השחק הטוב ביותר ועוד.

**הנחת עבודה:**

* עליכם לממש Attributes, Constructors, Getters & Setters, toString לפי הצורך.
* ניתן להוסיף מתודות שירות המסוגלות לספק חישובים הרלוונטיים לסטטיסטיקות.
* עליכם להשתמש בקבועים כשניתן.
* על הקוד להיות קריא מסודר ויעיל ככחול האפשר.
* אין צורך לבצע בקרת ערכים, אלא במקומות שנרשם מפורשות: גובה, גילאים וכו'.

**סעיף 1 – בניית מחלקות בסיסיות - 10 נקודות**

ממשו את המחלקות הבאות בהתאם למאפיינים המפורטים בהמשך.

* Run
  + Methods
    - + main(String[]args) : void
      * Will initialize an instance of League class.
* Person (abstract)
  + Attributes
    - -Id : int
      * static counter: 1,2,…,n (autoincrement)
    - -name : String
      * static counter: 1,2,…,n (autoincrement)
    - -age : int
      * Values: 18-35
    - -height : float
      * Values: 1.65 - 2.2 m
  + Constructors
    - + Person()
      * Generate random value inside the constructor.
  + Methods
    - +toString(): String

print e.g. “Id: 1, Name: name1, Age: 10, Height: 1.9 ”

* Couch (extends Person)
* Player (extends Person)
  + Attributes
    - -shotTwo : int[2]
      * [0] is Hit, [1] is Miss
    - -shotThree : int[2]
      * [0] is Hit, [1] is Miss
    - -shotRateTwo : final float
      * Random constant 75 - 90%
    - shotRateThree : final float
      * Random constant 25 - 40%
    - -totalScore : int
      * Both 2-point shots and 3-point shots
  + Constructors
    - + Player()
      * Generate random value inside the constructor.
  + Methods
    - +throwBallTwo : boolean
      * Formula to generate random shots:

0.1f + (new Random().nextFloat() \* 0.9f)

* + - * Return True if the random value is equal or higher than shotRateTwo  
        Else return false
    - +throwBallThree : boolean
      * Formula to generate random shots:

0.1f + (new Random().nextFloat() \* 0.9f)

* + - * Return True if the random value is equal or higher than shotRateThree  
        Else return false
    - +toString(): String
      * Make sure to use Super().toString

**סעיף 2 – בניית תת מחלקות של מחלקת Person - 20 נקודות**

ממשו את המחלקות הבאות בהתאם למאפיינים המפורטים בהמשך.

* Team
  + Attributes
    - -name : String
    - -players : Player[5]
    - -totalScore : int
    - - couch : Couch
  + Constructors
    - + Team()
      * Generate random value inside the constructor.
  + Methods
    - Play() : void
      * Let a random team player throw the ball with random throw between 2 point throw to 3 point throw
    - teamMVP() : void
      * print the most valuable player of the game.
      * MVP is the player with the most score in the game
    - toString(): String
* Game
  + Attributes
    - -gameNumber : int
      * static counter: 1,2,…,n (autoincrement)
    - -teams : Team[2]
  + Constructors
    - + Game()
      * Generate random value inside the constructor.
  + Methods
    - startGame() : void
      * Let a random team to start first and each team have 1 turn each time.
    - gameSummary() : void
      * print the summary of the game
      * for e.g:  
        Team 1: 59 | Team 2: 68  
        Team 2 won the game!  
        Team 2 MVP player is : player3

**סעיף 3 – בניית מחלקות מורכבות – 30 נקודות**

ממשו את המחלקות הבאות בהתאם למאפיינים המפורטים בהמשך.

* League
  + Attributes
    - name : String
    - games : Game[3]
    - wonTeams : Team[2]
  + Methods
    - startLeagueGames(): void
    - toString(): String
      * Use super toString if needed.

**סעיף 4 – בניית מחלקת Run - 20 נקודות**

צרו מחלקה בשם Run ובה מתודת ראשית main().

יש לייצר משחק בין 2 קבוצות ולהתחיל את המשחק

**סעיף 5 – בניית מחלקת עזר GameStatistics – 20 נקודות**

במחלקת GameStatistics יש ליצור 3 מתודות סטטיות שידפיסו את הנתונים הבאים

1. ממוצע ניקוד של כלל השחקנים במשחק (2 הקבוצות).
2. השחק בעל הניקוד הכי נמוך בליגה.

יש להיעזר בבניית מתודות תומכות במחלקות ולהקפיד על כתיבת קוד נכון.

**סעיף 6 בונוס – ניתן לבחור עד 3 סעיפים - סה"כ 6 נקודות מקסימליות**

1. ממוצע של גילאי השחקנים בקבוצה.
2. ממוצע גובה השחקנים בקבוצה.
3. השחקן בעל אחוזי הקליעה הגבוהים לסל של 2 נקודות
4. השחקן בעל אחוזי הקליעה הגבוהים לסל של 3 נקודות