



## הנדסת תוכנה – תרגיל בית 2

### דגשים להגשת המטלה

1. **תאריך הגשה: יום ראשון 01.01.2023 בשעה 23:59.**
2. **הגשה בזוגות בלבד!**
3. הקוד חייב להיכתב בהתאם למוסכמות כתיבת הקוד בקורס כולל תיעוד כנדרש. קוד שלא עומד בדרישות יגרור הורדת ניקוד. ניתן למצוא את קובץ מוסכמות הקידוד באתר הקורס תחת הלשונית "קבצי עזר".
4. ההגשה מתבצעת ב-Moodle באזור המיועד על ידי אחד מהשותפים, לאחר יצירת קבוצה.
5. ניתן להגיש את התרגיל לכל היותר עד 48 שעות לאחר מועד ההגשה ללא הורדת ניקוד. לאחר 48 שעות תיבת ההגשה תיסגר ולא יהיה ניתן להגיש את התרגיל כלל.
6. פורמט הגשת התרגיל נמצא בקובץ ההנחיות ב-Moodle. **כל חריגה מפורמט זה תגרור ציון אפס.**

### מטרת התרגיל

עבודה עם מחלקות והורשה.

### הכנות טרם תחילת התרגיל

1. פתיחת פרויקט ג'אווה חדש.
2. הורדת קבצי התרגיל, והעתקת הקובץ Main.java **בלבד** אל תוך תיקיית ה-src.

### הוראות כלליות

1. יש לבדוק שהקוד עובר הידור (קומפילציה) ללא שגיאות.
2. מומלץ להריץ את התוכנית עם מספר קלטים שונים ולחשוב על מקרי קצה אפשריים.
3. מומלץ לחזור על התרגולים וההרצאות וכן להיעזר באינטרנט.
4. יש להשתמש בגרסה 9.0.4 של ג'אווה בעת פתרון התרגיל.
5. מומלץ להשתמש ב-Git במהלך כתיבת התרגיל.

### הוראות הגשה

1. יש למלא את הוראות ההגשה בהתאם למסמך הדרישות "הנחיות כלליות לפתרון והגשת תרגילי הבית" אשר מופיע באתר הקורס.
2. הגשה אלקטרונית **בלבד** דרך אתר הקורס ב-moodle. ההגשה תכלול קובץ ה-zip בלבד בפורמט `HW2_<id1>_<id2>` כאשר `<id1>` ו-`<id2>` הם תעודות הזהות של המגישים. פירוט נוסף נמצא בסוף המסמך.
3. ההגשה מתבצעת על ידי אחד מבני הזוג לאחר שיצר קבוצה וכן הזוג השני הצטרף אליה.
4. תרגיל בית שלא יוגש על פי הוראות ההגשה – **לא ייבדק**.
5. יש להקפיד על יושרת הכנת התרגיל וההגשה.
6. יש לוודא כי הקוד מתקמפל – קוד אשר לא יעבור הידור יקבל ציון אפס.
7. אין צורך להגיש את קובץ הפלט אשר ניתן כחלק מתרגיל זה.



## חלק א – שוויון עצמים

בתרגול 4 דיברנו על שוויון עצמים, ובפרט על שוויון עצמים כאשר משתמשים בהורשה.

בחלק זה של התרגיל יהיה עליכם לממש את המחלקות Clock ו-AccurateClock אשר ראינו בתרגול, ולדרוס בהן את הפעולות equals, hashCode, toString.

על פעולות ה-equals וה-hashCode של המחלקות לעמוד במפרטים של פעולות ה-equals וה-hashCode כפי שהם מוגדרים במחלקה Object (בהתאמה).

בנוסף, על יחס השוויון לקיים מספר תכונות נוספות:

- כל שני מופעים של המחלקה Clock יהיו שווים זה לזה אם ורק אם ערכי השעות והדקות שלהם שווים זה לזה (בהתאמה).
- כל שני מופעים של המחלקה AccurateClock יהיו שווים זה לזה אם ורק אם ערכי השעות, הדקות והשניות שלהם שווים זה לזה (בהתאמה).
- לכל משתנה c בעל טיפוס דינמי Clock ולכל משתנה ac בעל טיפוס דינמי AccurateClock על הקריאות c.equals(ac) ו-ac.equals(c) להחזיר שקר (false), ללא תלות בערכי התכונות של המופעים.

את פעולת ה-toString במחלקה Clock יש לממש כך שהשעון יוצג בפורמט HH:MM (גם כאשר אחד המספרים הוא חד-ספרתי) ואת פעולת ה-toString במחלקה AccurateClock יש לממש כך שהשעון יוצג בפורמט HH:MM:SS (גם כאשר אחד המספרים הוא חד-ספרתי). בעת מימוש הפעולה במחלקה AccurateClock יש להיעזר בפעולת ה-toString של המחלקה Clock.

## הוראות נוספות:

- בכתיבת פעולות ה-equals חל איסור על שימוש בפעולות אחרות המוגדרות במחלקה Object, ובפרט בפעולה getClass.
- יש לממש את פעולות ה-hashCode באופן בו לכל שני מופעים של אותה המחלקה אשר אינם שווים זה לזה יהיה ערך גיבוב (hash value) שונה.
- במחלקה Clock אין לאזכר את המחלקה AccurateClock, ובמחלקה AccurateClock אין לאזכר את המחלקה Clock פרט להגדרת ההורשה בכותרת הפעולה (כלומר, חל איסור שבקוד של מחלקה אחת יופיע השם של המחלקה השנייה, פרט לשורה בה מגדירים כי המחלקה AccurateClock יורשת מן המחלקה Clock).
- בכל מקרה בו מנסים להגדיר לתכונה כלשהי ערך שאינו בטווח החוקי לתכונה זו יש לשנות את הערך של התכונה לערך ברירת המחדל 0.



## חלק ב – ביטויים אריתמטיים

בחלק זה של התרגיל תבנו מערכת לייצוג ביטויים אריתמטיים.

### סיפור המערכת

ביטוי אריתמטי הינו ביטוי חשבוני המורכב ממספרים ומפעולות חשבון.

המערכת אותה תבנו מורכבת מעשרה סוגים של ביטויים אריתמטיים:

- מספר שלם: ביטוי אריתמטי אשר מורכב ממספר שלם יחיד.
- מספר ממשי: ביטוי אריתמטי אשר מורכב ממספר ממשי יחיד.
- חיבור: ביטוי אריתמטי אשר מורכב מסכום של שני ביטויים אריתמטיים.
- מולטי-חיבור: ביטוי אריתמטי אשר מורכב מסכום של לפחות שני ביטויים אריתמטיים.
- חיסור: ביטוי אריתמטי אשר מורכב מהפרש של שני ביטויים אריתמטיים.
- כפל: ביטוי אריתמטי אשר מורכב ממכפלה של שני ביטויים אריתמטיים.
- מולטי-כפל: ביטוי אריתמטי אשר מורכב ממכפלה של לפחות שני ביטויים אריתמטיים.
- חילוק: ביטוי אריתמטי אשר מורכב ממנה של שני ביטויים אריתמטיים.
- מינוס אונארי: ביטוי אריתמטי אשר מורכב מביטוי אריתמטי אחר, ומייצג את הביטוי המנוגד לו.
- עיגול: ביטוי אריתמטי אשר מורכב מביטוי אריתמטי אחר, ומעגל את ערכו למספר נתון של ספרות אחרי הנקודה.

כל ביטוי אריתמטי ניתן לחישוב ולהצגה:

- חישוב: כאשר מחשבים ביטוי אריתמטי, התוצאה המתקבלת היא ערכו המספרי של הביטוי.
- הצגה: כאשר מציגים ביטוי אריתמטי, התוצאה המתקבלת הינה מחרוזת המציגה את הביטוי לפי הביטויים המרכיבים אותו, מוקף בסוגריים עגולים משני הצדדים.

לדוגמה, נתבונן בביטוי  $3 * (1 + 2)$ . ערכו של ביטוי זה הינו 9, והצגתו הינה  $((1 + 2) * (3))$ .

### מימוש המערכת

המערכת אותה תבנו תתמוך בביטויים אריתמטיים מסוגים שונים.

על מנת לממש את המערכת, תצטרכו ליצור מספר מחלקות. מחלקות אלו יתבססו אחת על השנייה, וביחד יהוו את מכלול המערכת.

### מחלקת Expression

מחלקה זו הינה מחלקה אשר מכילה שתי פעולות מופשטות:

- פעולה `double evaluate()` אשר מחזירה את ערכו של הביטוי.
- פעולה `String toString()` אשר מחזירה מחרוזת המייצגת את הביטוי.



#### IntegerLiteral מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג מספר שלם.

#### DoubleLiteral מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג מספר ממשי.

#### Addition מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג חיבור.

#### MultiAddition מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג מולטי-חיבור.

#### Subtraction מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג חיסור.

#### Multiplication מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג כפל.

#### MultiMultiplication מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג מולטי-כפל.

#### Division מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג חילוק.

במחלקה זו יש עליכם לוודא כי לא מתבצעת חלוקה באפס. במידה ואכן מתבצעת חלוקה כזו, עליכם לזרוק חריגה בלתי מסומנת בשם `DivisionByZeroException` בעת חישוב ערך הביטוי. חריגה זו אינה מוגדרת בג'אווה ועליכם להגדירה בעצמכם. במחלקה יש להגדיר את שלושת הבנאים הבסיסיים.

#### UnaryMinus מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג מינוס אונארי.

#### RoundedExpression מחלקת

מחלקה זו מייצגת ביטוי אריתמטי מסוג עיגול. הצגתו של ביטוי אריתמטי מסוג זה מוגדרת להיות הצגה של הביטוי המרכיב את ביטוי העיגול.



## הנחיות לפתרון

- לפני פתרון התרגיל, התחילו בלכתכנן את היחסים והקשרים בין המחלקות השונות.
- בעת פתרון התרגיל ניתן ואף מומלץ להגדיר מחלקות נוספות על מנת לחסוך בשכפול קוד כמה שניתן.
- בעת פתרון התרגיל, יש להקפיד על שמות משמעותיים למשתנים, לפעולות ולמחלקות.
- בעת פתרון התרגיל, יש להקפיד על הרשאות הגישה השונות בהתאם לנלמד בקורס.
- בעת פתרון התרגיל, חשבו האם ישנן מחלקות אשר אמורות להיות מופשטות.
- יש ליצור כל מחלקה בקובץ נפרד.
- אין לייבא בקוד ספריות או חבילות כלשהן.
- בכל מחלקה יש לכלול את התכונות המתאימות לה, ולספק בעבורן פעולות `get` ו-`set` במידת הצורך בלבד.
- על המחלקות להכיל בנאים בהתאם לצורך. אין חובה שבנאי יקבל כפרמטרים את כל התכונות אשר מוגדרות במחלקה (ואף זה אינו רצוי בעבור חלק מן המחלקות).
- יש לתעד את כל הפעולות והמחלקות אותן אתם מגדירים בעזרת שימוש ב-Javadoc בהתאם לקובץ מוסכמות התיעוד אשר מופיע באתר הקורס. על בתיעוד לכלול פסקת `throws` בפעולות הרלוונטיות. בנוסף, יש לתעד שורות קוד אשר עשויות להיות קשות להבנה.
- בעת פתרון התרגיל ניתן ואף מומלץ להגדיר קבועים ולא להשתמש במספרי קסם.
- בכל דריסה ומימוש של פעולה יש להשתמש באנוטציה `@override` על מנת לוודא שאכן מתבצעת דריסה.
- יש לדאוג לכך שניסיון יצירה של ביטויים מסוג מולטי-חיבור ומולטי-כפל עם פחות משני פרמטרים יוביל לשגיאת הידור.

## הגשת התרגיל

לפני הגשת התרגיל, עליכם ליצור קבצי `html` מתוך תיעוד ה-Javadoc שכתבתם בקוד. ניתן ליצור את הקבצים בעזרת ה-IntelliJ על ידי לחיצה על כרטיסיית ה-Tools ובחירת האפשרות `Generate Javadoc`. בעת יצירת המסמכים, יש לבחור את הרשאות הגישה `private` על מנת שכל המחלקות והפעולות (משני חלקי התרגיל) יופיעו במסמכים. את כל הקבצים שנוצרים יש לשמור בתיקייה בשם `Javadoc` אשר ממוקמת בתוך תיקיית ה-`src`.

כפי שהוזכר, קובץ ההגשה הסופי הינו קובץ `zip` בודד. שם הקובץ מפורט בתחילת המסמך ובמסמך ההוראות הכללי. קובץ ה-`zip` יכלול תיקייה בודדת בשם `src` בה יימצאו כל קבצי הקוד אותם כתבתם ותיקייה נוספת בשם `Javadoc` אשר תכיל את כל קבצי התיעוד.

## הרצת התוכנית וביצוע בדיקות

במחלקה `Main` קיימות מספר פעולות אשר משמשות לבדיקת הקוד. להזכירכם, חלק מן הבדיקה נעשה באופן אוטומטי, ולכן אין לשנות את המחלקה `Main` ואת הפעולה הראשית, ובפרט אין לשנות את פעולות ההדפסה המתבצעות בה.

על מנת לוודא כי הקוד שכתבתם עובד כראוי, מצורף לתרגיל זה קובץ הפלט `HW2_output.txt`, על מנת שתוכלו לבצע את ההשוואה באופן ידני (או על ידי שימוש ב-DiffMerge).

בהצלחה