

עבודה בית 1

מבוא ובקרת זרימה

הנחיות:

- העבודה להגשה עד התאריך 3/4/2025 בשעה 23:50. שימו לב התיבה נסגרת בשעה המצוינת ולכן לא יהיה ניתן להגיש את העבודה באיחור. אל תחכו לרגע האחרון!
- רק במקרים חריגים תתאפשר הגשה באיחור.
- קראו היטב את השאלות. עליכם לענות על כל השאלות בעבודה זו. כל השאלות בעבודה בעלות אותו משקל. כלומר הציון מתחלק שווה בשווה בין כל השאלות בעבודה.
- הגשת העבודה תהיה דרך אתר הקורס בלי-מוד בלבד. הגשת העבודה היא **ביחידים**.
- ההגשה של העבודה תתבצע באופן הבא:
 - עבור כל שאלה יש ליצור קובץ java עם סיומת .java. השמות של הקבצים יהיו ex1.java, ex2.java, ex3.java, ex4.java,...
 - חובה לכווץ את כל הקבצים יחדיו לקובץ אחד ויחיד בפורמט RAR או ZIP, ולהגיש רק אותו.
 - יש להגיש רק תוכניות שעוברות קומפילציה תקינה ללא שגיאות על קומפיילר שפת java. קבצים שלא מתקמפלים יגררו ציון 0 עבור אותה השאלה.
- בתחילת כל קובץ יש להוסיף את התיעוד הבא. יש לשנות את השם לשם שלכם ואת תעודת הזהות לתעודת הזהות שלכם.

// Assignment: 1

// Author: Israel Israeli, ID: 01234567

- לאחר שהעליתם את הקבצים שלכם ללי-מוד, הורידו אותם מהמודל למחשב שלכם וודאו כי הקבצים תקינים וכי העליתם את הקבצים הנכונים והמלאים. לאחר תום מועד ההגשה לא יתקבלו ערעורים על כך שהעליתם קבצים לא תקינים או שהעליתם בטעות קבצים אחרים / לא נכונים.
- שאלות בנוגע העבודה יש לשאול באימייל או בשעות הקבלה. שאלות על העבודה לא יענו בזמן ההרצאה או המעבדה. אימייל: yaelva@mail.sapir.ac.il
- ניתן להשתמש בכל החומר שנלמד עד לולאות (כולל). אסור להשתמש בחומר שטרם נלמד!
- יש להקפיד על תכנות נכון. יש לכתוב **הערות באנגלית בלבד**, יש להקפיד על הזחות, כיתוב נכון/קריא, שמות משמעותיים וכיוצא באלו.

בהצלחה!

משימה 1: שער חליפי

כתבו תכנית הקולטת מספר ממשי המייצג שער מטבע (\$) וסכום בדולרים ומדפיסה את הסכום במונחי שקל חדש (₪) לפי השער.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע אדום הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

Enter exchange rate (USD to NIS): 3.5

Enter amount in USD: 100

The amount in NIS is: 350.00

משימה 2: המרת מרחקים

כתבו תכנית הקולטת מספר ממשי המייצג מרחק בקילו-מטרים (ק"מ), ומדפיסה את המרחק במייל. מייל אחד שווה 1.609 קילומטר.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע אדום הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

Enter distance in kilometers: 20

The distance in miles is: 12.4274 miles

משימה 3: סוגי משולשים

כתבו תוכנית מבקשת כקלט אורכים של שלוש צלעות של משולש וקולטת אותם כמספרים שלמים (int).

1. התוכנית בודקת שני סוגים של טעויות בקלט:

- התוכנית בודקת שאורכי כל הצלעות חיוביים. במידה ולא, התוכנית תדפיס: "Error" ותסיים

- התוכנית בודקת האם ניתן לייצר משולש על ידי האורכים שהתקבלו. נבדוק אם סכום האורכים של כל שתי צלעות גדול מהשלישי. במידה ולא, התוכנית תדפיס:

“ We cannot make a triangle from these edges. ”

ותסיים

2. אם הקלט תקין ואכן ניתן ליצור משולש מהצלעות הללו יש להדפיס את סוג המשולש הכי מתאים מבין הסוגים הבאים:

- א. משולש שונה צלעות (כל הצלעות שונות) – Scalene triangle.
- ב. משולש שווה שוקיים (לפחות שתי צלעות שוות) – isosceles triangle.
- ג. משולש שווה צלעות (כל הצלעות שוות) – equilateral triangle.

שימו לב: כי משולש שווה צלעות הוא גם שווה שוקיים, אך במקרה זה נבחר בסוג שווה צלעות.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע **אדום** הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

דוגמא:

Enter lengths of three edges: -10 20 50

Error

דוגמא:

Enter lengths of three edges: 10 20 50

We cannot make a triangle from these edges.

דוגמא:

Enter lengths of three edges: 30 20 50

We cannot make a triangle from these edges.

דוגמא:

Enter lengths of three edges: 30.5 20 50

Scalene triangle

דוגמא:

Enter lengths of three edges: 40 40 50

Isosceles triangle

דוגמא:

Enter lengths of three edges: 50 50 50

Equilateral triangle

משימה 4: איסוף שטרות ומטבעות.

כתבו תוכנית שתקלוט מהמשתמש מספר תלת ספרתי (מספר שלם).

1. התוכנית תבדוק שהמספר תקין כלומר המשתמש הכניס מספר תלת ספרתי. אחרת, היא תדפיס למסך "Error" ותסיים.
2. התוכנית תציג את כמות השטרות (שטרות של 200,100,50 או 20) והמטבעות (מטבעות של 10,5,2 או שקל) שיהיה עלינו לאסוף על מנת להרכיב את המספר שנקלט.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע **אדום** הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

דוגמא:

Enter a three-digit number: **258**

1 Note(s) of 200.0
0 Note(s) of 100.0
1 Note(s) of 50.0
0 Note(s) of 20.0
0 Note(s) of 10.0
1 Note(s) of 5.0
1 Note(s) of 2.0
1 Note(s) of 1.0

דוגמא:

Enter a three-digit number: **42**

Error

משימה 5: הדפסת שעות פעילות.

כתוב תוכנית שקולטת יום בשבוע (מספר בין 1-7) ומדפיסה את שעות פעילות המוקד הטלפוני באותו היום.

המוקד יהיה פתוח בשעות הבאות:

ביום א', ג', ד', ו-ה': 10:00-13:00

ביום ב': 9:00-14:00

ביום ו': 10:00-12:00

בנוסף, ביום ד' המוקד יהיה פתוח: 16:00-19:00

ביום שבת: המוקד סגור.

- שימו לב עליכם לממש את התוכנית בעזרת שימוש בswitch.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע **אדום** הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

דוגמא:

Enter a number representing the day of the week (1-7): **1**

The call center is open from 10:00 to 13:00.

דוגמא:

Enter a number representing the day of the week (1-7): **4**

The call center is open from 10:00 to 13:00.

Additionally, the call center is open from 16:00 to 19:00.

דוגמא:

Enter a number representing the day of the week (1-7): **7**

The call center is closed.

משימה 6: מספרים מספרות זהות

כתבו תוכנית מבקשת כקלט שני מספרים שלמים שונים.

התוכנית בודקת ששני המספרים שונים זה מזה, אחרת תבקש מהמשתמש להכניס קלט עד אשר הקלטו שני מספרים שלמים שונים זה מזה. בכל פעם שהקלט לא תקין תצא הודעת שגיאה למשתמש ובקשה לקלט חוזר.

שימו לב: המספרים לא חייבים להיות חיוביים.

אם הקלט תקין התוכנית תבדוק אם המספר מורכב מספרות זהות. כמות הפעם שכל ספרה חוזרת על עצמה במספר היא חסרת משמעות.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע **אדום** הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

דוגמא:

Enter two different integers: **15474 7451**

Both numbers are composed of identical digits.

דוגמא:

Enter two different integers: 784 674

The numbers are composed of different digits.

דוגמא:

Enter two different integers: 10 10

Error: The numbers must be different. Please enter two different numbers.

Enter two different integers: -46784 7684

Both numbers are composed of identical digits.

משימה 7: מספרים סימטריים.

מספר סימטרי (פלינדרום) הוא מספר שלם שקריאתו משמאל לימין זהה לקריאתו מימין לשמאל.

לדוגמה: 98189, 54745, 1221

כתבו תוכנית אשר מבצעת את הפעולות הבאות:

- מבקשת וקולטת מהמשתמש מספר שלם חיובי n מטיפוס `int`.
- בודקת תקינות הקלט:
 - א. כי הקלט חיובי והמספר מקיים את התנאי הבא: $1 \leq n \leq 9$
 - ב. במידה והקלט אינו תקין התוכנית תחזיר הודעת שגיאה ותבקש להזין את שני המספרים שוב. עד אשר הקלט יהיה תקין.
- מדפיסה את כל המספרים שהם מספרים סימטריים בין 10^n ל- 10^{n+1} .

אין להשתמש בתוכנית זו בשיטות קיימות ב-`java`.

להלן דוגמאות של הרצת התוכנית. בדוגמאות, מה שבצבע אדום הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

דוגמא:

Enter a positive integer n: 2

Palindrome numbers between 100 and 999:

101, 111, 121, 131, 141, 151, 161, 171, 181, 191, 202, 212, 222, 232, 242, 252, 262, 272, 282, 292, 303, 313, 323, 333, 343, 353, 363, 373, 383, 393, 404, 414, 424, 434, 444, 454, 464, 474, 484, 494, 505, 515, 525, 535, 545, 555, 565, 575, 585, 595, 606, 616, 626, 636, 646, 656, 666, 676, 686, 696, 707, 717, 727, 737,

747, 757, 767, 777, 787, 797, 808, 818, 828, 838, 848, 858, 868, 878, 888, 898, 909, 919, 929, 939, 949, 959, 969, 979, 989, 999

דוגמא:

Enter a positive integer n: -3

Error: Please enter a positive integer greater than 0.

משימה 8: מספרי ארמסטרונג

מספר ארמסטרונג (או מספר נרקסיסטי) הוא מספר שלם N שלם שמקיים את התכונות הבאות:

- $N > 0$.
- המספר הוא בעל n ספרות. נסמן את הספרות משמאל לימין: a_1, a_2, \dots, a_n .
- כך ש- $N = (a_1)^n + (a_2)^n + (a_3)^n + \dots + (a_n)^n$.

דוגמאות:

153 הוא מספר ארמסטרונג $153 = 1^3 + 5^3 + 3^3$

1634 הוא מספר ארמסטרונג $1634 = 1^4 + 6^4 + 3^4 + 4^4$

371 הוא מספר ארמסטרונג $371 = 3^3 + 7^3 + 1^3$

כתבו תוכנית אשר מבצעת את הפעולות הבאות:

4. מבקשת וקולטת מהמשתמש שני שלמים אי שליליים (גדולים או שווים לאפס) מטיפוס `int`: `lowerBound` ו-`upperBound`.

5. בודקת תקינות הקלט:

א. כי הקלט אינו שלילי.

ב. כי $\text{lowerBound} \leq \text{upperBound}$.

ג. במידה והקלט אינו תקין התוכנית תחזיר הודעת שגיאה ותבקש להזין את שני המספרים שוב.

ד. יש שלשה נסיונות לקבלת קלט תקין. לדוגמא אם הנסיונות לקלט היו:

• 10,-5

• 10,0

• 10,100

אז התוכנית תפסיק לקבל קלט ותוציא הודעה שגיאה ותסיים.

6. מדפיסה את כל המספרים שהם מספרי ארמסטרונג בתחום מ-`lowerBound` (כולל) ועד `upperBound` (כולל).

7. אם אין מספרי ארמסטרונג בתחום התוכנית תוציא הודעה

No Armstrong numbers in this range

אין להשתמש בתוכנית זו בשיטות קיימות ב-java להעלאה בחזקה (אשר אותן עדיין לא למדנו).

להלן דוגמא של הרצת התוכנית. בדוגמאות, מה שבצבע אדום הוקלד ע"י המשתמש. בשחור - הדפסות של התוכנית.

דוגמא:

Enter the lower and upper bound for numbers: 0 50654

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 153, 370, 371, 407, 1634, 8208, 9474

דוגמא:

Enter the lower and upper bound for numbers: 370 100000

370, 371, 407, 1634, 8208, 9474, 54748, 92727, 93084

דוגמא:

Enter the lower and upper bound for numbers: 666666 999999

No Armstrong numbers in this range

דוגמא:

Enter the lower and upper bound for numbers: 0 -5

Error, please try again

בהצלחה!