**public class** Die {

**private int** numOfSides;

**private int**[] statistics;

**public** Die(){

**this**(6);

**שאלה 1 (מתייחסת לתוכנית שמדמה זריקת קוביה בהרצאת מערכים)**

}

**public** Die(**int** n) { numOfSides=n;

statistics=**new int**[numOfSides];

}

**public int** rollDie(){

**int** result; result=1+(**int**)(Math.*random*()\*numOfSides); statistics[result-1]++;

**return** result;

}

**public void** showStatistics(){

**int** k;

**for** (k=0; k<numOfSides ; k++) System.***out***.println(k+1 + ":" + statistics[k]);

הוסיפו פה את הקוד לסעיף 1.2 //

}

}

ולהלן תוכנית Program שמבצעת זריקת קוביה 1000 פעמים ומציגה סטטיסטיקות מזריקות אלו

**public class** Program {

**public static void** main(String[] args) {

Die my=**new** Die();

**for** (**int** k=1;k<=1000;k++) my.rollDie(); my.showStatistics();

}

}

1.1 השיטה הנתונה rollDie מגרילה ומחזירה מספר בטווח 1 ל .numOfSides הוסיפו לשיטה פרמטר חדש בוליאני

בשם .unique אם הארגומנט ש- unique מקבל הוא false השיטה תיפעל כפי שהיא כתובה כעת ואין צורך בשינוי, אם הפרמטר יקבל true השיטה תחזיר ערך אקראי שונה מההגרלה האחרונה שהייתה. כלומר, אם בקריאה הקודמת הערך שהוחזר היה ,5 השיטה תחזיר 1,2,3,4 או 6 אך לא .5 לביצוע המשימה על השיטה לזכור מה היה ערך

ההגרלה האחרונה כדי להשוות אותו עם ערך ההגרלה. רמז: הגדירו תכונה חדשה בשם lastToss שתשמור את ערך

ההגרלה האחרונה. עליכם לשנות את הקריאות לשיטה rollDie במקומות האחרים בתוכנית.

1.2 הוסף למחלקה ,Die שיטה פרטית getMostFrequent שתחזיר כערך החזרה את הספרה השכיחה ביותר (זוהי

הספרה עבורה המונה המתאים במערך הסטטיסטיקות הוא הגבוהה ביותר.)

1.3 בסוף השיטה showStatistics הוסף קוד שיקרא לשיטה getMostFrequent וידפיס את השורה :

"Most frequent digit is " + (getMostFrequent - ל מהקריאה המוחזר הערך)

אם יש מספר ספרות כאלו יש להחזיר אחת מהן. להלן דוגמא לפלט אפשרי של התוכנית :

1:179

2:154

3:179

4:172

5:164

6:152

Most frequent digit is 3

(שימו לב כי במקרה זה גם הספרה 1 וגם הספרה 3 הן ספרות שכיחות ביותר ולכן גם השורה האחרונה הבאה הייתה

נכונה במקום השורה הנוכחית:

Most frequent digit is 1

1.4 הוסף למחלקה ,Die שיטה ציבורית getAllMostFrequents שתחזיר כערך החזרה מערך עם כל הספרות השכיחות ביותר. עבור מערך הסטטיסטיקות שהוצג קודם – על השיטה להחזיר את המערך

במידה וקיימת ספרה אחת שכיחה ביותר יש להחזיר אותה במערך בגודל .1

|  |  |
| --- | --- |
| 1 | 3 |

1.5 שנו את השיטה הראשית main שבמחלקה Program כך שתבצע :

א) להדפיס את השורה :

System.***out***.println("Please enter number of sides:");

ב) לקלוט מספר s מהמשתמש (על ידי שימוש ב)Scanner-

ג) להדפיס את השורה :

System.***out***.println("Please enter number of rolls in the experiment:");

ד) לקלוט מספר r מהמשתמש (על ידי שימוש ב)Scanner-

ה) לאחר מכן, על השיטה לבצע בלולאה :

.A להדפיס את השורה :

מספר האיטרציה + " # iteration "Starting

בכל פעם)

ומתקדם ב1-

(מספר האיטרציה מתחיל ב1-

.B יצירת אובייקט חדש מטיפוס Die בעל s צדדים.

.C ביצוע r פעמים rollDie() על האובייקט שיצרתם.

.D ביצוע showStatistics() על האובייקט שיצרתם.

.E בדיקה האם יש יותר מספרה אחת שכיחה ביותר. אם כן, יש לצאת מן הלולאה.

ו) לאחר היציאה מהלולאה יש להדפיס את השורה :

"Found multiple max frequency. Experiment ended after " +

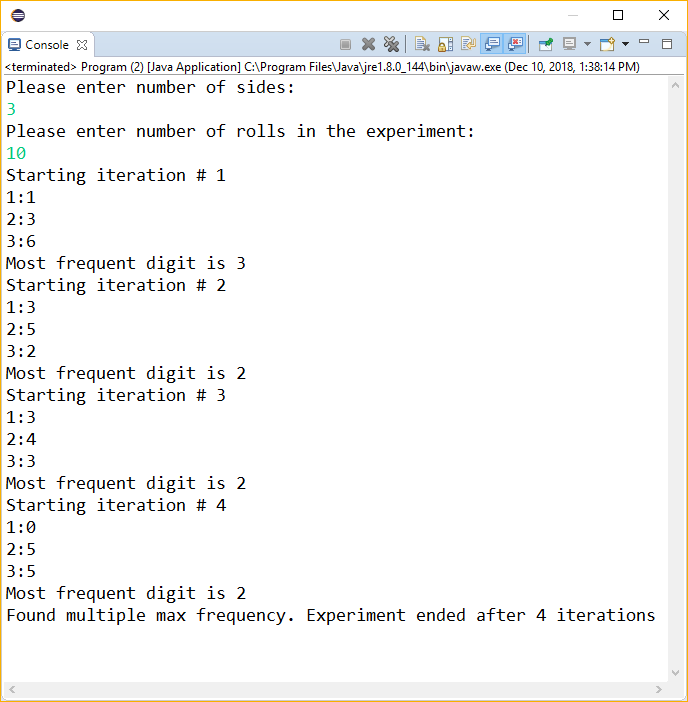
+ " iterations");

משמאל

מספר האיטרציות שהתבצעו

דוגמא לפלט אפשרי של התוכנית.

בהרצה זו המשתמש בחר להשתמש בDie- עם 3 צדדים (קוביה שמכליה רק את הספרות



)1,2,3 ובכל פעם להטיל את הקוביה 10 פעמים.

הלולאה הסתיימה לאחר 4 איטרציות .

(באיטרציה האחרונה הספרות 2,3 הן כולן

"שכיחות ביותר)"

בכל האיטרציות לפני כן רק ספרה אחת היא

"שכיחה ביותר."

Set המתארת קבוצה של מספרים שלמים. לדוגמא, תהיינה s1 וs2- הקבוצות: 6 , 1 , 3 , 2 : s1

s2 : 12 , 7 , 2 , 1 , 8 , -1 , 0 ,3

**שאלה 2**

בשאלה זו תכתבו מחלקה בשם

שימו לב:

 כל האיברים בקבוצה שונים זה מזה.

 אין חשיבות לסדר האיברים בתוך הקבוצה.

אוביקט חדש מטיפוס Set ייתאר קבוצה ריקה, כאשר בהמשך, נאפשר הוספה והוצאה של מספרים מהקבוצה. לצורך אחסון המספרים נשתמש במערך. לפיכך, גודל המערך יהיה "הגודל המקסימלי של הקבוצה" – כמות האיברים המקסימלית שיכולה להיות בקבוצה. מספר זה יועבר כפראמטר לבנאי של .Set שימו לב להבדל בין "הגודל המקסימלי של הקבוצה" לבין "הגודל בפועל" של הקבוצה – כמות האיברים בקבוצה ברגע מסוים (מספר זה כמובן משתנה לאורך ביצוע

התוכנית כאשר איברים מוכנסים ומוצאים מהקבוצה.)

לדוגמא:, אם הגודל המקסימלי של הקבוצה s1 הוא (נניח) 10 וכיון שהגודל בפועל של s1 הוא ,4 אז ניתן להוסיף לs1- עוד 6

איברים נוספים לפני שהקבוצה תתמלא לגודלה המקסימלי. ואם הגודל המקסימלי של הקבוצה s2 הוא (נניח) 8 וכיון שהגודל בפועל של s2 הוא כאמור 8 ,אז לא ניתן להוסיף לs2-

עוד איברים נוספים כלל – לפחות עד שיוצא מs2- אחד מהאיברים ש s2- מכילה כרגע.

2.1 הגדירו את המחלקה Set והוסיפו את התכונות שיאפשרו שמירת נתוני הקבוצה כפי שתוארה. (רמז – אחת מהתכונות

היא כאמור מטיפוס "מערך של מספרים שלמים" - int[] .)

2.2 הגדירו בנאי ציבורי שיאפשר יצירת קבוצה חדשה. על הבנאי לקבל כפראמטר את הגודל המקסימלי של הקבוצה.

כאשר הקבוצה נוצרת מתקיים:

 הבנאי יקצה את הקבוצה בגודלה המקסימלי

 הקבוצה הנוצרת ריקה. (כל התאים שהוקצו לא מכילים מספרים , אינם מלאים.) צריך להגדיר תכונה, ,counter

שתפקידה לציין כמה תאים בתוך הקבוצה מלאים. בעת יצירתה של הקבוצה תכונה זו שווה 0 כי הקבוצה ריקה.

וערך התכונה counter יהיה .0 : s1

למשל כאשר ניצור קבוצה חדשה בשם Set(10) new = s1 היא תראה כך

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |

2.3 הגדירו שיטה ציבורית בשם add שמקבלת כפרמטר מספר שלם ומוסיפה אותו לקבוצה. הפונקציה תחזיר כערך החזרה

(כלומר השיטה מחזירה ערך מטיפוס

אחרת

במידה והצליחה להוסיף את המספר לקבוצה וfalse-

true

את הערך

.)boolean השיטה מחזירה false אם אין יותר מקום בקבוצה או שהאיבר כבר קיים בה.

s1:

s1 אם נוסיף את המספר 7 ( נבצע s1.add(7) ) היא תראה כך :

למשל, עבור הקבוצה

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 |  |  |  |  |  |  |  |  |  |

כרגע, אפשר להגיד ש- s1 היא קבוצה שגודלה הוא 1 כי היא מכילה איבר אחד )7( ולכל היותר ניתן למלא אותה ב- 10

איברים. שימו לב שהערך 7 התווסף למקום counter=0 ומיד אחרי ההוספה הערך של counter יקודם ל- .1

ואם נוסיף את המספר 1 הקבוצה תראה כך:

s1:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | 1 |  |  |  |  |  |  |  |  |

וכעת, הקבוצה s1 מכילה שני איברים וגודלה הוא 2 ולכל היותר ניתן למלא בה 10 איברים.

אם ננסה להוסיף את המספר 7 שוב פעם, הוא לא יתווסף כי הוא קיים כבר והקריאה s1.add(7) תחזיר ערך .false

2.4 הגדירו שיטה ציבורית בשם find שמקבלת כפרמטר מספר שלם ובודקת אם הוא קיים בקבוצה.

הפונקציה תחזיר כערך החזרה את אינדקס המספר במערך במידה והמספר שנשלח אליה כפראמטר קיים בקבוצה ו- -1

אחרת .

s1 (שכבר יש בה שני איברים בשלב הזה:)

- תחזיר ערך .0

למשל, עבור הקבוצה הקריאה s1.find(7)

הקריאה s1.find(10) - תחזיר ערך .-1

)value,value,value,…(

show שמדפיסה את הקבוצה בפורמט הבא:

2.5 הגדירו שיטה ציבורית בשם

לדוגמא,הקריאה s1.show() תדפיס את השורה הבאה.

(7,1)

( פתיחת סוגריים ומיד סגירת הסוגריים)

()

show() בקבוצה ריקה תדפיס את השורה הבאה.

הקריאה

השיטה show() אינה מחזירה ערך כלשהוא.

2.6 הגדירו שיטה בשם isEmpty שמחזירה אמת )true( אם הSet- ריקה ושקר (false) אחרת. כלומר, שיטה זו תחזיר

אמת אם אין מספרים בקבוצה בכלל.

2.7 הגדירו שיטה בשם isFull שמחזירה אמת אם הקבוצה מלאה ושקר אחרת. השיטה תחזיר אמת אם כמות האיברים ב-

Set שווה לגודלו המקסימלי.

2.8 הגדירו שיטה בשם remove אשר מקבלת מספר שלם כלשהוא כפראמטר ומנסה למחוק אותו מהקבוצה. מחיקת

המספר משמעותה הזזת כל איברי המערך צעד אחד שמאלה ודריסת הערך המבוקש. למשל, אם נבקש למחוק את המספר

8 מהקבוצה הבאה:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | 1 | 2 | 8 | 15 | 6 | 7 |  |  |  |

הקבוצה תראה כך לאחר המחיקה:

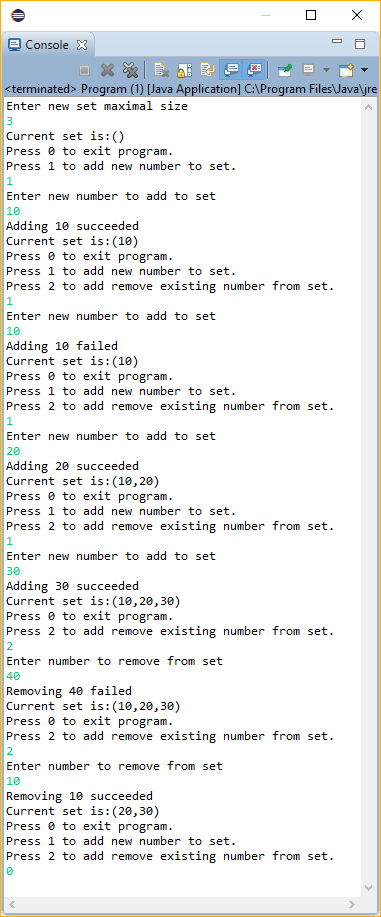
|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | 1 | 2 | 15 | 6 | 7 |  |  |  |  |

אם המספר שהועבר כפראמטר כלל אינו בקבוצה – אין לעשות דבר עם המספרים שבSet- ולהחזיר .false

ניתן להשתמש בשיטה find כדי למצוא את ממיקום הערך במערך ומשם לבצע הזזה שמאלה כדי למחוק אותו.

שימו לב שיש לעדכן את ערך התכונה counter לאחר המחיקה.

2.9 בשיטה הראשית main שבמחלקה Program תכתבו תוכנית שמאפשרת למשתמש "לתחזק" קבוצה Set שכזה.

באופן מדויק, על השיטה:

"Enter new set maximal size" :להדפיס )1

)2 לקלוט מספר מהמשתמש )3 ליצור אוביקט מסוג קבוצה בגודל מכסימלי שהכניס המשתמש

)4 לבצע בלולאה

.a הדפסה (בשורה אחת) את המילים is:" set "Current

ולאחריה קריאה ל.show()-

"Press 0 to exit program." השורה הדפסת .b

.c אם הקבוצה לא מלאה יש להדפיס השורה:

"Press 1 to add new number to set."

.d אם הקבוצה לא ריקה יש להדפיס השורה:

"Press 2 to remove existing number from set."

.e לקלוט מספר מהמשתמש

.f אם המשתמש לחץ 0 – יש לצאת מהלולאה ולסיים התוכנית.

.g אחרת, אם המשתמש לחץ 1 על התוכנית:

"Enter new number to add to set"

.i להדפיס

לקלוט מספר מהמשתמש לנסות להכניסו לקבוצה.

על התוכנית לדווח אם הצליחה ההכנסה או לא.

.ii

.iii

.h אחרת, אם המשתמש לחץ 2 על התוכנית:

"Enter number to remove from set"

.i להדפיס

לקלוט מספר מהמשתמש לנסות להוציאו מהקבוצה.

על התוכנית לדווח אם הצליחה ההוצאה או לא.

.ii

.iii

משמאל דוגמא הרצה מצופה. – אתם יכולים לראות שם את אופן

הדיווח המצופה בתתי סעיפים iii g ו- iii .h

בדוגמא זו המשתמש : יצר קבוצה חדשה בגודל מקסימלי 3

הוסיף לה את האיבר 10 (הוספה שהצליחה)

הוסיף לה את האיבר 10 (הוספה שנכשלה)

הוסיף לה את האיבר 20 (הוספה שהצליחה)

הוסיף לה את האיבר 30 (הוספה שהצליחה) הוציא ממנה את האיבר 40 (הוצאה שנכשלה כמובן כי

40 אינו איבר בקבוצה)

הוציא ממנה את האיבר 10 (הוצאה שהצליחה)

יצא מהלולאה (לחץ )0

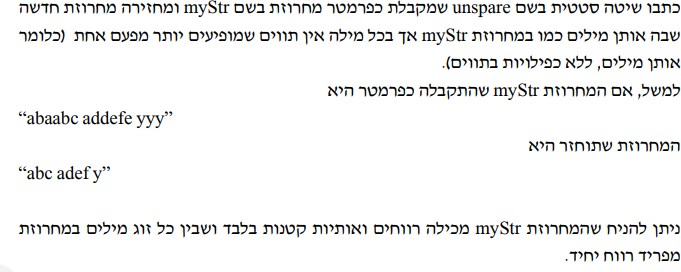
חשוב מאד : יש להקפיד לבצע את הקוד Scanner(System.in) new = בדיוק פעם אחת בלבד (הכי פשוט בתחילת

השיטה ואח"כ להמשיך להשתמש באותו אוביקט Scanner כדי לבצע את כל הnextInt()- שאתם צריכים).

**שאלה 3 - בשאלה זו תכתבו רק את המחלקה .Program**

3.1 הוסיפו למחלקה Program את השיטה הסטטית unspare המתוארת להלן (מתוך מועד א תשע"ח.)

ניתן ואף מומלץ (אבל לא חובה) להוסיף שיטות עזר שיעזרו לכם לפתור את הבעיה.



3.2 כתוב במחלקה Program שיטה סטטית ציבורית (נוסף ל)main- בשם reverse שמקבלת כפרמטר מחרוזת

(שמכילה אותיות ורווחים בלבד) ומחזירה מחרוזת חדשה שמכילה את מילות המחרוזת המקורית באותו הסדר אך עם

.“sihT si a tset”

test a is "This תוחזר המחרוזת

תווים הפוכיל בכל מילה. למשל עבור המחרוזת "

שים לב שהרווחים מצומצמים במחרוזת החדשה.

לדוגמא,

System.***out***.println(*reverse*("This is a test ")); // " sihT si a tset" ידפיס System.***out***.println(*reverse*("walla yofi ")); // "allaw ifoy" ידפיס System.***out***.println(*reverse*(" w a ll a y ofi")); // "w a ll a y ifo" ידפיס System.***out***.println(*reverse*("")); // ריקה מחרוזת ידפיס

System.***out***.println(*reverse*(" w ")); // "w" ידפיס

3.3 בשיטה הראשית main שבמחלקה Program כיתבו קוד ש:

)1 הדפסת השורה Start” “The

)2 מבצע בלולאה (אינסופית, היציאה מהלולאה בתת-סעיף :)ii

.a קבלת מחרוזת מהמשתמש (על ידי שימוש בשיטה nextLine() של המחלקה )Scanner

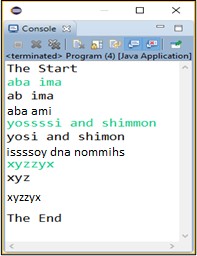
.b אם המחרוזת ריקה יש לצאת מהלולאה. .c הדפסת הערך המוחזר מהשיטה *unspare* שמופעלת על

המחרוזת שנקלטה מהמשתמש.

.d הדפסת הערך המוחזר מהשיטה *reverse* שמופעלת על

המחרוזת שנקלטה מהמשתמש.

)3 הדפסת השורה End” “The

חשוב מאד ! יש לבצע במהלך התוכנית רק פעם אחת את הקוד

**new** Scanner(System.***in***);

ובפרט לא להכניס את השורה הזו לתוך הלולאה. (זה יפיל את

הבדיקה) ניתן ואף חובה להכניס את הקריאה nextLine() לתוך הלולאה (כפי

שהתבקשתם בסעיף 2 .)a משמאל דוגמת הרצה של התוכנית. (זיכרו כי השורות הירוקות הן

הקלטים אותם הכניס המשתמש). לפני שורת הפלט End The ישנה שורה ריקה (שורת קלט) שהכניס

המשתמש.