

**מס' נבחן**

**שם הקורס: מבוא לתכנות ג'אבה – פתרון !!!**

**קוד הקורס: 833428211**

|  |  |
| --- | --- |
| **הוראות לנבחן:**   * **חומר עזר שימושי לבחינה:** * **כל חומר עזר מודפס** * **מחשבונים** | **בחינת סמסטר: קיץ**  **השנה: תשע"ח**  **מועד: ב** |
| * **אין לכתוב בעפרון / עט מחיק** * **אין להשתמש בטלפון סלולארי** * **אין להשתמש במחשב אישי או נייד** * **אין להשתמש בדיסק און קי ו/או   מכשיר מדיה אחר** * **אין להפריד את דפי שאלון הבחינה** | **תאריך הבחינה:**  **שעת הבחינה:**  **משך הבחינה: 180 דקות**  **מרצים: אפי פרופוס** |

**בהצלחה!**

כל הזכויות שמורות ©. מבלי לפגוע באמור לעיל, אין להעתיק, לצלם, להקליט, לשדר, לאחסן מאגר מידע, בכל דרך שהיא, בין מכאנית ובין אלקטרונית או בכל דרך אחרת כל חלק שהוא מטופס הבחינה

|  |
| --- |
| **\*\*\* שאלון הבחינה לא ייבדק ע"י המרצה, לא ייסרק ולא יישמר \*\*\***  **\*\*\* לא יינתן ציון על תשובות אשר תיכתבנה בשאלון זה \*\*\***  **מבנה הבחינה והנחיות לפתרון:**   * במבחן 4 שאלות, יש לענות על כולן. משקל כל שאלה מופיע בגוף הבחינה. * על הקוד להיות פשוט, מסודר וברור. קוד מסורבל יגרור הורדת ניקוד. * יש להשתמש בפונקציות שנלמדו, ואין "להמציא את הגלגל מחדש", אלא אם נאמר במפורש אחרת * **אין להשתמש בחומר שלא נלמד בקורס זה** * יש להקפיד על כל כללי הכתיבה שנלמדו, ובפרט על קריאות, מודלוריות, שימוש בקבועים ופשטות. * מותר ומומלץ לכתוב פונקציות עזר במקרה הצורך!  שאלה 1 (20 נקודות) כתבו מחלקה Cat המייצגת חתול. למחלקה תהיה מספר שלם המתאר את משקל החתול, מחרוזת המתארת את שם החתול, מספר שלם המתאר את מספר השבב של החתול, ומספר שלם המתאר מחיר החתול.  יש לממש למחלקה:   1. בנאי המקבל את כל הפרמטרים לשדות. ניתן להניח שהקלט תקין. 2. מתודות get/set מתאימות עבור כל הפרמטרים. 3. מתודה boolean isLower(int x) המקבלת ארגומנט x ומחזירה true אם גודלו של החתול קטן מ x. במקרה אחר – המתודה תחזיר false. 4. מתודה boolean isItforMe(int x,int y) אשר מחזירה true במידה ומחיר החתול נמוך מערך y וגם מתודת ה isLower עם המשתנה x שכתבתם בסעיף קודם (3) מחזירה true . אחרת המתודה תחזיר false   **public** **abstract class** Cat {  **private** **int** catW;  **private** String name;  **private** **int** id;  **private** **int** price;      **public** Cat(**int** catW, String name, **int** id, **int** price) {    **this**.catW = catW;  **this**.name = name;  **this**.id = id;  **this**.price = price;  }  **public** **int** getCatW() {  **return** catW;  }  **public** **void** setCatW(**int** catW) {  **this**.catW = catW;  }  **public** String getName() {  **return** name;  }  **public** **void** setName(String name) {  **this**.name = name;  }  **public** **int** getId() {  **return** id;  }  **public** **void** setId(**int** id) {  **this**.id = id;  }  **public** **int** getPrice() {  **return** price;  }  **public** **void** setPrice(**int** price) {  **this**.price = price;  }  **public** **boolean** isLower(**int** x)  {  **Return** (catW > x)    }  **boolean** isItforMe(**int** x,**int** y)  {  **Return** (price < y && isLower(x))    }      } שאלה 2: (20 נקודות) בהמשך לשאלה 1: מחלקת חתול סיאמי (SiameseCat) הינו סוג של חתול, ויורש ממחלקת Cat, מחלקת SiameseCat מכיל גם מחרוזת המתארת את גוון האפור של החתול, ישנם 3 דרגות אפור: אפור כהה ( dark gray ), אפור רגיל ( regular gray ) ואפור בהיר ( baby gray ). יש לממש עבור מחלקת חתול סיאמי:   1. בנאי המקבל את כל הפרמטרים לשדות. אין צורך לבדוק את תקינות הקלט. 2. מתודות get/set מתאימות עבור כל הפרמטרים. 3. מתודה הדורסת את toString() עבור חתוך סיאמי כך שתחזיר מחרוזת ובה כל נתוני חתול סיאמי ( משקל החתול, שם החתול, מספר השבב של החתול, מחיר החתול, גוון האפור של החתול ). 4. מתודה הדורסת את equals של Object – שני חתולים סיאמים ייחשבו שווים (המתודה equals תחזיר true) אם גוון האפור שלהם שווה וגם מספר השבב שלהם שווה.   פתרון:  **public** **class** SiameseCat **extends** Cat{  **private** String grayRate;    **public** SiameseCat(**int** catW, String name, **int** id, **int** price, String grayRate) {  **super**(catW, name, id, price);  **this**.grayRate = grayRate;  }    **public** String getGrayRate() {  **return** grayRate;  }  **public** **void** setGrayRate(String grayRate) {  **this**.grayRate = grayRate;  }  @Override  **public** String toString()  {  **return** getName()+" "+ getId()+" "+getCatW()+ " "+getPrice()+" "+ grayRate;    }  @Override  **public** **boolean** equals(Object o)  {  **if**(o **instanceof** SiameseCat)  {  SiameseCat f = (SiameseCat) o;  **if**(f.getGrayRate() == grayRate && f.getId() == getId())  {  **return** **true**;  }  }    **return** **false**;  }    } שאלה 3: (20 נקודות) בהמשך לשאלה 1: מחלקת חתול פרסי (PersianCat) הינו סוג של חתול, ויורש ממחלקת Cat, והוא בנוסף מכיל גם ערך נוסף הנקרא מדד החתול האופטימלי – ערכי המדד הם : a עבור חתול ברמה נמוכה, b עבור חתול ברמה בינונית ו c עבור חתול ברמה גבוהה. יש לממש עבור מחלקת חתול פרסי:   1. בנאי המקבל את כל הפרמטרים לשדות. אין צורך לבדוק את תקינות הקלט. 2. מתודות get/set מתאימות עבור כל הפרמטרים מלבד מדד החתול. 3. מתודת מדד החתול void rateTheCat() אשר תקבע את רמת החתול לפי הנוסחא הבאה:   (מחיר החתול כפול ( 2 ספרות אחרות של מספר השבב בחתול ) ) חלקי 30 .  במידה והתוצאה קטנה מ 200 החתול ידורג החתול ברמה נמוכה (a) במידה והתוצאה בין 200 ל 1000 החתול ידורג ברמה בינונית (b ) ובמידה והתוצאה גבוהה מ 1000 החתול ידורג ברמה גבוהה (c ).   1. מתודה הדורסת את toString() עבור חתול פרסי כך שתחזיר מחרוזת ובה כל נתוני חתול פרסי (משקל החתול, שם החתול, מספר השבב של החתול, מחיר החתול, ורמת החתול ) 2. מתודה הדורסת את equals של Object – שני חתולים פרסים יהיו שווים אם מדד הדירוג שלהם זהה.   פתרון:  **public** **class** PersianCat **extends** Cat{  **private** **char** rateCat;    **public** PersianCat(**int** catW, String name, **int** id, **int** price, **char** rateCat) {  **super**(catW, name, id, price);  **this**.rateCat = rateCat;  }    **public** **char** getRateCat() {  **return** rateCat;  }  **public** **void** rateTheCat()  {  **double** rate = (**double**) ((getPrice() \* getId()%100) /30);    **if**(rate < 200)  rateCat = 'a';  **else** **if**(rate>200 && rate <1000)  rateCat = 'b';  **else**  rateCat = 'c';      }      @Override  **public** String toString()  {  **return** getName() + " " + getId() + " "+ getCatW() + " "+ getPrice()+ " "+ rateCat;    }  @Override  **public** **boolean** equals(Object o)  {  **if**(o **instanceof** PersianCat)  {  PersianCat f = (PersianCat) o;  **if**(f.getRateCat() == rateCat)  {  **return** **true**;  }  }    **return** **false**;  }    } שאלה 4: (40 נקודות) בהמשך לשאלות 1-3: כתבו מחלקה המתארת חנות Store ובה מערך חתולים למכירה. חתול יכול להיות חתול סיאמי או חתול פרסי.   1. יש לממש בנאי (ללא פרמטרים). כשמאתחלים את החנות אין בה עדיין חתולים. 2. צרו פונקציה menu() int אשר תאפשר למשתמש לבחור סוג של החתול. הפונקציה תחזיר את מספר הבחירה של המשתמש.   לדוגמא:  MENU  Pick your choice:  [1] – Siamese Cat  [2] – Persian Cat  [3] – exit   1. יש לממש מתודה להוספת חתול לחנות addCat() void המקבלת את כל הפרמטרים הנדרשים – בכפוף לסוג החתול שנבחר על ידי המשתמש בסעיף קודם (2), ומוסיפה אובייקט מתאים בסוף מערך החתולים שבחנות.   שימו לב כי על המערך להיות דינאמי וגודלו יתאר במדויק את כמות החתולים שקיימים בו ( כמות החתולים בחנות ).   1. יש לממש מתודה deleteCat (String name) void המקבלת שם של מודל ומוחקת ממערך החתולים את כל החתולים בעלי אותו שבב ( שימו לב כי לכל חתול ישנו שבב יחודי שלא חוזר על עצמו ). 2. באמצעות השיטות שלמדתם בכיתה, מיינו את מערך החתולים בחנות לפי מחירם מהנמוך לגבוה. ( אין להשתמש בפונקציות קיימות של המערכת ל collections ). 3. יש לממש מתודה void printBestCat (int rate) אשר עוברת על מערך החתולים שבחנות וכותבת לקובץ output.txt את כל פרטיהם של החתולים הפרסים בעלי דירוג c או חתולים סיאמים בעלי צבע אפור כהה, שמחירם ( נכון לגבי שני סוגי החתולים ) הוא מעל ערך הארגומנט rate. במקרה ואין חתולים שעונים לתיאור – יש לכתוב הודעה מתאימה.   עליכם להשתמש בפונקציות שכבר כתבתם בשאלות קודמות כמו tostring , ובמבנה של האלגוריתם לחיפוש שנלמד בכיתה ( שימו לב שבסעיף 5 כבר מיינתם את החתולים בסדר עולה ).  פתרון:  import java.io.FileWriter;  import java.io.IOException;  import java.io.PrintWriter;  import java.util.ArrayList;  import java.util.Scanner;  public class Store {  private static FileWriter fw;  private static PrintWriter pw;  private ArrayList<Cat> arr;    public Store()  {  arr = new ArrayList<Cat>();  try {    fw = new FileWriter("out.txt" );  pw = new PrintWriter(fw);    } catch (IOException e) {  // TODO Auto-generated catch block  e.printStackTrace();  }    }    public int menu()  {  Scanner input = new Scanner(System.in);  System.out.println("MENU");  System.out.println("Pick your choice");  System.out.println("[1] - Siamese Cat");  System.out.println("[2] - Persian Cat ");  System.out.println("[3] - exit");  return input.nextInt();    }  public void addCat(int catW , String name , int id, int price, grayRate )  {  arr.add(new SiameseCat (catW , name , id, price, String grayRate));  }  public void addCat(int catW , String name , int id, int price, char rateCat)  {  arr.add(new PersianCat (catW , name , id, price, rateCat));  }  public void deleteCat(int id)  {  for(int i=0 ; i< arr.size() ; i++)  {  if(arr.get(i).getId() == id)  {  arr.remove(i);  i--;  }  }  }  public void BubbleSort()  {    int n = arr.size();  Cat temp ;  for(int i=0; i < n; i++){  for(int j=1; j < (n-i); j++){  if(arr.get(j-1).getPrice() > arr.get(j).getPrice()){  //swap elements  temp = arr.get(j-1);  arr.set(j-1, arr.get(j)) ;  arr.set(j, temp);  }  }  }    }  public void printBestAquarium (int rate)  {  for(Cat a: arr)  {  if ( a instanceof SiameseCat)  {  SiameseCat s = (SiameseCat) a;  if(s.getGrayRate().equals("darkGray"))  pw.write(a.toString());    }  if ( a instanceof PersianCat)  {  PersianCat s = (PersianCat) a;  if(s.getRateCat() == 'c' )  pw.write(a.toString());    }    }    }    }  **בהצלחה !** |