

# המסלול האקדמי המכללה למינהל

## ביה"ס למדעי המחשב



ת.ז. הסטודנט: \_\_\_\_\_  
מספר חדר: \_\_\_\_\_  
מספר נבחן: \_\_\_\_\_  
מספר אסמכתא: \_\_\_\_\_

ברקוד נבחן

### מבחן בקורס: פיתוח תוכנה מתקדם 1

תאריך הבחינה: 10.09.17

שנת הלימודים: תשע"ז, סמסטר: א', מועד: מיוחד

משך הבחינה: 3 שעות

שם המתרגל/ים:

רועי יהושוע

שם המרצה/ים:

ד"ר אליהו חלסצ'י

מבנה הבחינה: הבחינה מורכבת מחלק אחד.

מספר השאלות הכולל בבחינה: 3.

משקל כל שאלה: בצמוד לכל שאלה

הוראות לנבחן:

- מותר השימוש בכל חומר עזר לא אלקטרוני
- יש לענות במחשב.
- לא נדרש להחזיר את השאלון.
- לא מצורף נספח לבחינה
- מחברת טיוטה : לא
- מחברת נפרדת לכל שאלה : לא

**בהצלחה!!**

## הקדמה

במבחן זה עליכם לענות על 3 \ 3 שאלות תכנותיות ב JAVA. משך המבחן 3 שעות. חומר פתוח. עליכם להקפיד היטב על ההוראות, ובפרט על הוראות ההגשה, שכן הבדיקה הינה אוטומטית.

אתם מקבלים:

- את קובצי המקור אותם עליכם להשלים
- MainAPI לבדיקה לוקאלית של ה API
- MainTrain של מוד האימון במערכת ההגשה עליכם להגיש
- את קובצי המקור מושלמים. לא ב zip או דומיו, אלא את קובצי המקור עצמם.

קוד שלא מתקמפל או שיש לו שגיאות בזמן ריצה יקבל הפחתה אוטומטית של עד 20 נקודות באופן יחסי לסעיף הבדיקה שנכשלה (וייבדק ידנית)

עליכם להקפיד על ה API הנדרש, הבדיקות של ה Main הלוקאלי, וכמובן על הוראות התרגיל, שכן ה Main הלוקאלי MainTrain בפירוש לא בודקים את כל המקרים שכן נבדקים במערכת האוטומטית של המבחן.

במהלך המבחן תוכלו להגיש את הקוד כמה פעמים שתמצאו גם במוד אימון וגם במוד הגשה. המבחן ייבדק רק ע"פ הקוד שהוגש למוד הגשה לפני שתם זמן המבחן. אל תטעו בהגשה או במודים. כל פרטי ההגשה נמצאים בסוף מסמך זה.

**אזהרה: כבר נתפסו מעתיקים בעבר. נא להימנע מדבר שעלול להרוס לכם את התואר.**

**הערה חשובה:** תקראו את טקסט השאלות של המבחן. כל המידע שאתם צריכים נמצא שם. כך תחסכו זמן ושאלות מיותרות.

**בהצלחה!**

אלי.

## שאלה 1 (35 נק'): מבני נתונים, תכנות גנרי

בתוך הקובץ Q1.java תמצאו את המחלקה `MyStack<T>` שצריכה לממש מחסנית גנרית. את המחסנית יש לממש באמצעות הרשימה המקושרת `list`. על המחסנית לממש את הפעולות הבאות:

- `push(T t)` דחיפת איבר לראש המחסנית
- `pop()` שליפת האיבר שבראש המחסנית
- `peek()` הצצה באיבר שנמצא בראש המחסנית (ללא הוצאתו)
- `contains(T t)` `boolean` המחזירה אמת במידה והאיבר נמצא במחסנית

תנאי הכרחי: על כל המתודות לעיל לפעול ב $O(1)$ זמן. כולל <code>contains</code> .
---

הערות:

- בהחלט ייתכן שנכניס למחסנית כמה פעמים את אותו האיבר
  - מותר לכם להוסיף מתודות ומשתי עזר כרצונכם.
- בסוף המחלקה תוכלו למצוא את המתודה `testAPI` הבודקת את ה `API` של המחלקה `MyStack`. בקובץ `MainTrain.java` תוכלו לראות שהטסט הראשון בודק האם `contains` אכן עובד ב  $O(1)$ . הטסט החסוי בודק אלמנטים הדומים לבדיקות ב `testAPI` של Q1.

## שאלה 2 (35 נק'): Object Adapter ,Decorator

בקובץ Q2.java נתונים הממשקים `StringReader` ו `IntArrayReader`. עליכם להשלים את המחלקות הבאות:

- המחלקה `MyWordFilter` שבהתאם לתבנית `Decorator`, עליה להיות `StringReader`, כך שתהפוך כל קריאה של מחרוזת למחרוזת בה כל מילה מופיעה בדיוק פעם אחת, לפי סדר הופעתה, ומופרדת ע"י ";". ניתן להניח כי המילים מופרדות ע"י רווח אחד.
  - לדוגמא הביטוי במתודה `testAPI` שנמצאת בסוף Q2. ניתן לראות כיצד עבור "Hello Hello World World Q2" נקבל בקריאה "Hello;World;Q2;"
- המחלקה `MyIntArrayToStringReader` שבהתאם לתבנית `Object Adapter` תבצע אדפטציה מ `IntArrayReader` ל `StringReader`, כך שבהינתן מערך של `int`-ים היא תחזיר מחרוזת בה כל `int` מהווה מילה מופרדת ע"י רווח.
  - ב `testAPI` ניתן לראות שעבור המערך 100,100,50,50,13,14 באמצעות ה `object adapter` וה `decorator` לעיל נקבל פלט של "100;50;13;14;"
  - הסבר: ה `adapter` הפך את המערך למחרוזת של מילים מופרדות ע"י רווח, וה `decorator` החזיר מחרוזת בה כל מילה מופיעה פעם אחת, לפי סדר הופעתה, מופרדת ע"י ";".

### שאלה 3 (30 נק'): כללית

במחלקה Q3 תמצאו את המתודות  $a()$ ,  $b()$ ,  $c()$ . בהתאם לסעיפים הבאים א - ג החזירו במתודות אלו `true` במידה והכתוב בסעיפים נכון, אחרת החזירו `false`. כל סעיף שווה 10 נקודות.

- בתכנות GUI נעדיף עורך ויזואלי שמייצר לנו את קוד המקור הנדרש בשפה בה אנו עובדים.
- העיקרון של `pure fabrication` אומר שאם אין אף מחלקה שמתאימה לקחת אחריות פונקציונאלית כלשהי, אז יש ליצור לשם כך מחלקה אבסטרקטית (`pure`).
- העיקרון של `Dependency Inversion` אומר שכדאי לנו להחזיק משתנים מהטיפוס האבסטרקטי ביותר במקום סוג קונקרטי כלשהו.

PTM1  
moed\_c  
הגשה  
#####  
Choose Files 4 files  
שלח

קורס  
שם המטלה  
אימון  
תז (9 ספרות):  
קבצים:

Welcome  
submit # 1/100000  
deleting old files...  
submitting...  
File Name: Q1.java  
File Name: Q2.java  
File Name: Q3.java  
File Name: Q4.java  
copying test file MainTest.java  
copying test file MainTrain.java  
compiling... running...  
your code is ok, no compilation or runtime errors. your grade will be published in a few days. good luck! :)

הגשה:

- <http://ck.cs.colman.ac.il>
  - כל המחלקות צריכות להיות ב package בשם `test`
  - עליכם להגיש למערכת ההגשה, תחת **PTM1** ו **moed\_c**, את הקבצים `Q1.java`, `Q2.java`, `Q3.java`, `Q4.java` בלבד.
  - כל הגשה צריכה לכלול את כל הקבצים האלה.
  - תבדילו בין מוד אימון למוד הגשה. רק מוד הגשה נחשב כהגשת המבחן.
  - המערכת לא תקבל ZIP או דומיו, או כל קובץ שאינו קוד מקור.
  - אנא אל תעשו טעויות מביכות שיעלו לכם במבחן. ההוראות ברורות ופשוטות.
- בנוסף הגישו את קוד המקור לשרת הגיבוי. כאן ניתן להשתמש ב `ZIP`. אך אנא וודאו שהוא מכיל את קוד המקור ושהוא לא יוצא לכם פגום. תבדקו את הקובץ לפני שליחה.

<http://db.cs.colman.ac.il/test/>

שרת הגיבוי מחזיר לכם מספר אסמכתא. נא להגיש אותה לבוחנות בסוף המבחן.