

תכנות מתקדם 2: 89-211 – מבחן לדוגמא

זמן המבחן: **שעתיים**, יש לענות על 6 מתוך 6 שאלות, **בגוף השאלון בלבד**. חומר סגור.

בקיאות

שאלה 1: (16 נק')

אנו בתהליך כתיבתה של אפליקציית desktop עבור מעה"פ חלונות. עלינו לחשוב קדימה כי ייתכן ונצטרך ליצור גרסת web בהמשך. לכן אליס מציעה שאת הקוד נחלק ל MVP, כאשר ב M "נספק את הסחורה" באמצעות תקשורת ל webservice שיבצע בפועל את החישוב. בוב מציע פשוט שיש לחלק את הקוד ל MVVM. מתוך הנחה שבבוא העת פרויקט ה web יהיה מבוסס multi-tier בתצורת MVC, מהם היתרונות והחסרונות של כל שיטה שהציעו אליס ובוב? נמקו

אליס, יתרונות (4 נק'): _____

חסרונות (4 נק'): _____

בוב, יתרונות (4 נק'): _____

חסרונות (4 נק'): _____

שאלה 2: (12 נק')

הקיפו בעיגול את התשובות הנכונות:

- סקלביליות ליניארית בהכרח דורשת יחס קבוע בין כמות העבודה לכמות המשאבים הנדרשים להתמודדות עמה.
- Bridge Design Pattern מאפשר לנו ליצור הפרדה בין האלגוריתם לבין הבעיה שאותה הוא פותר.
- הטלת אחריות על מחלקה כלשהי צריכה להתבצע ע"פ רלוונטיות האינפורמציה שמחלקה זו מכילה; זה אחד מהעקרונות של Solid.
- Duplicateability היא אחד מהעקרונות של ACID.

מיומנות עיצוב קוד (Design) וכתיבת קוד

שאלה 3: (20 נק') ברצוננו ליצור web service המקיים את הממשק MazeGenerator שהגדיר את המתודה `Maze generateMaze(int rows, int cols);`. ביצענו חקר ביצועים עבור שלושה אלגוריתמים שונים A, B, C. גילנו שמתחת ל 30 תאים עדיף להשתמש ב A, בין 30-100 תאים עדיף להשתמש ב B, ומעל 100 תאים עדיף להשתמש ב C. ייתכנו בעתיד מימושים נוספים לאלגוריתמים חדשים שנבחר בהם ע"פ מדיניות שתיקבע ע"י חקר ביצועים.

עליכם לשרטט תרשים מחלקות (class diagram) ב UML, המציג עיצוב מונחה עצמים שמאפשר את מימוש ה webservice לעיל, ושומר על עקרון ה open\close.

תשובה: (על השרטוט להכיל את כל המחלקות הרלוונטיות + הערות ע"פ הצורך)

תשובה:

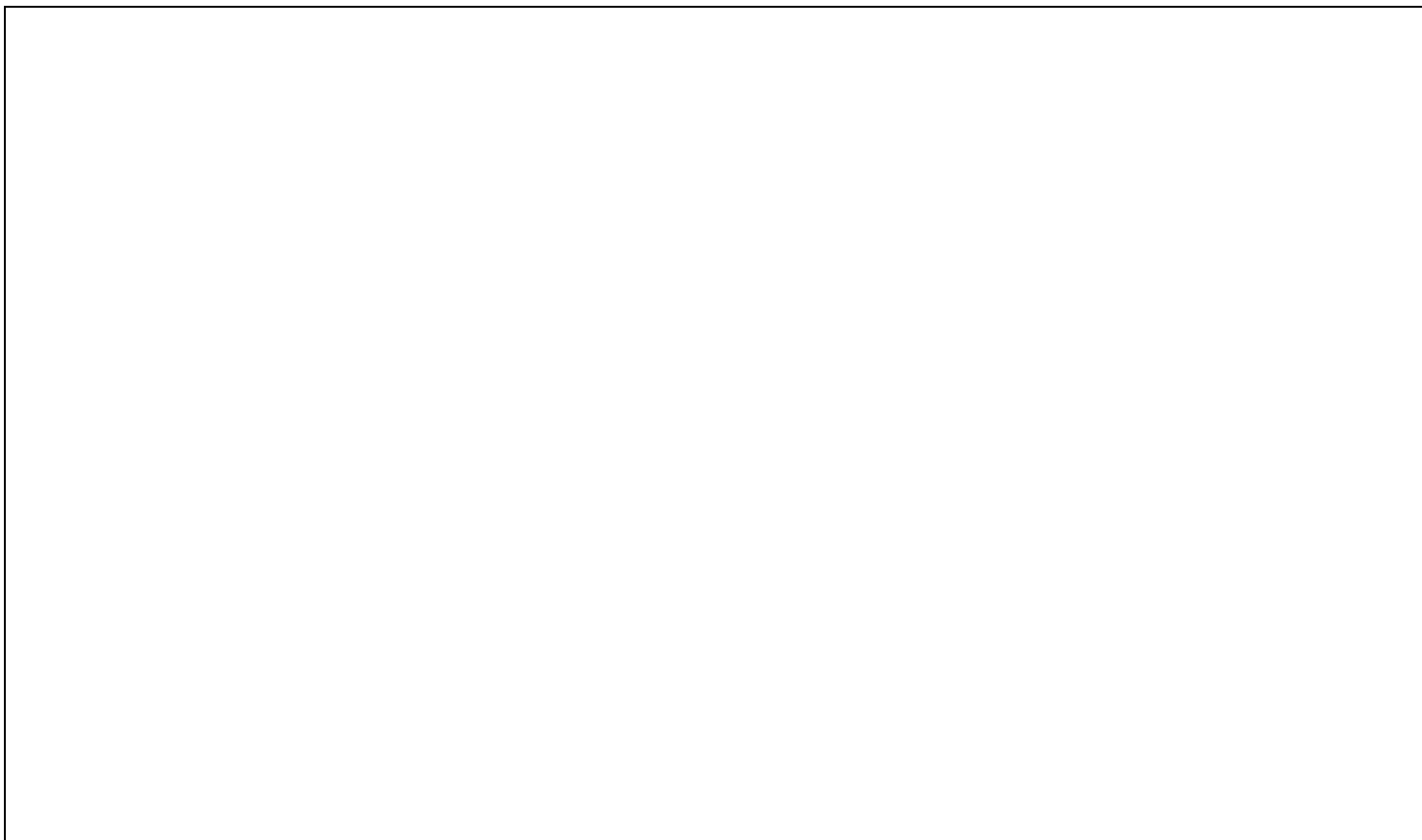
[illegible]

This image shows a full page of blank white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page, providing a template for writing or drawing. There are no margins, text, or other markings on the paper.

שאלה 5: (20 נקודות)

עבור הבעיה המוצגת בשאלה 4, עליכם לשרטט תרשים מחלקות (class diagram) ב UML, שעושה שימוש ב service locator כ design pattern כדי לפתור את הבעיה. (במידה ומימשתם service locator בשאלה 4, ממשו כאן פתרון שעונה על החסרונות של service locator)

תשובה:



שאלה 6: ממשו בקוד (באיזו שפה מונחית עצמים שתמצו) את מחלקת ה service locator עבור הבעיה לעיל.

בהצלחה!

תשובה 6:

[illegible]