



מכון טכנולוגי חולון
Holon Institute of Technology

מטלה מסכמת - מבוא למדעי המחשב

משחק 4 בשורה

מטרת המטלה

המטרה העיקרית: לפתח משחק אינטראקטיבי "ארבע בשורה" הכולל מספר מצבי משחק ורמות קושי שונות. המערכת תפותח בשפת C בשילוב AI.

המטלה נועדה לתרגל תהליך של תכנון, בניית פתרון ועבודה עצמאית (למידה עצמית) תוך הקפדה על עקרונות תכנות בסיסיים וכתובת קוד קריא ומודולרי.

דרישות כלליות למטלה:

- שימוש בפונקציות ופירוק הקוד ליחידות לוגיות
- שימוש במערכים דו-ממדיים
- שימוש באלגוריתמים בסיסיים
- תפריט משתמש ברור וידידותי

חויית משתמש נוחה וברורה

העבודה תבוצע בזוגות (ניתן גם להגיש לבד)

תיאור כללי

התוכנית תדמה משחק **ארבע בשורה (Connect Four)** על

לוח בגודל 7×6 . (<https://bit.ly/4a4tmbg>)

לכל שחקן יש דסקיות בצבע ייחודי. כל שחקן בתורו בוחר עמודה ומטיל לתוכה דסקית, אשר נופלת למיקום הפנוי הנמוך ביותר בעמודה. המנצח הוא השחקן הראשון שמצליח ליצור רצף של ארבע דסקיות מאותו הצבע ברצף אופקי, אנכי או אלכסוני.

אם הלוח מתמלא לחלוטין ואין רצף של ארבע דסקיות מאותו הצבע, המשחק מסתיים בתיקו.

התוכנית תכלול **תפריט** ראשי שיאפשר בחירה בין מצבי המשחק הבאים:

- משתמש מול משתמש
- משתמש מול מחשב
- צפייה בסטטיסטיקות משחקים
- יציאה מהתוכנית



במצב "משתמש מול מחשב" – יינתן למשתמש לבחור בין כמה רמות קושי:

- קל - מהלכים פשוטים
- בינוני - ניסיונות חכמים בסיסיים
- קשה (אופציונלי) - אסטרטגיה מתקדמת יותר

דרישות המערכת

על המערכת לתמוך ביכולות הבאות:

- אתחול לוח משחק
- הצגת הלוח (לימוד ושימוש בגרפיקה מתקדמת – אופציונלי)
- מהלך שחקן (הכנסת דסקית לעמודה חוקית, טיפול בקלט לא תקין, החלפת תורות)
- בדיקות ניצחות (בדיקה האם יש 4 בשורה, בעמודה או באלכסון)
- בדיקת תיקו (לא נשאר מקום בלוח)
- תפריט ראשי: שחקן נגד שחקן, שחקן נגד מחשב, יציאה
- משחק מול מחשב – התוכנית תאפשר למשתמש לבחור רמה מתוך לפחות 2 דרגות: רמה פשוטה ורמה יותר חכמה.
- בונוס: ממשק גרפי/ויזואלי/ ASCII פשוט

דרישות תכנותיות

דרישות חובה:

- פירוק לפונקציות - אין לכתוב את כל הקוד ב- main. יש לחלק את הפונקציונליות לקבצי H ו C כלפי הצורך.
- קריאות הקוד - שמות משתנים ופונקציות ברורים באנגלית, תיאור קצר וברור של תפקיד כל פונקציה מרכזית, הזחה נכונה, הערות במקומות המתאימים (כמות מינימלית)
- קלות הפעלה - תפריט פשוט עם אפשרויות ממוספרות, טיפול נבון בקלט לא תקין

כללי שימוש ב - AI

מותר ורצוי להשתמש ב: AI

- כלים מומלצים ChatGPT, Copilot, Gemini ועוד:
- למטרות: קבלת רעיונות למבני נתונים, הסברים על שפת C, עזרה באיתור שגיאות (debugging) עצות לשיפור מבנה הקוד ועוד.

חובות חשובות:

- קובץ תיעוד שמושים ב AI: חובה לצרף קובץ שמכיל:
 - איזה כלים השתמשתם - ChatGPT, Copilot וכו.
 - דוגמאות בולטות לשאלות/פניות שנתתם לכלי
 - אילו חלקים בקוד קיבלו השראה או נכתבו בעזרת הכלי

- הבנה עמוקה של הקוד: חובה להבין את הקוד לעומק – שלא יתברר בבחינה בעל-פה שאינכם מבינים חלקים כגון קוד שנלקח מ AI ללא הבנה

הגשה ובדיקה

אופן ההגשה:

- הגשה בזוגות - כל זוג יגיש פרויקט אחד
- ההגשה במודל (Moodle) (על כל סטודנט להגיש את המטלה במודל על אף שהעבודה בזוגות)
- פורמט ההגשה: קובץ ZIP המכיל את כל הפרויקט (קובץ SLN וכל קבצי H I C)
- מצגת חובה - הסבר קצר על הפונקציות העיקריות, והסבר קצר על רמות קושי אילו הרחבות עשיתם (אם יש)
- קובץ תיעוד שימוש ב AI

בחינה בעל-פה:

לאחר ההגשה יתקיים מבחן בעל-פה. כל אחד מבני הזוג יישאל על חלקים שונים בקוד. כל סטודנט/ית יקבל/תקבל ציון נפרד בהתאם להבנה שהפגין/ה.

קריטריונים להערכה:

הציון הסופי יבנה מהמשקל הבא:

- 25% תקינות פונקציונלית - האם כל הדרישות הבסיסיות עובדות
- 15% קריאות וארגון הקוד - מודולריות, פירוק לקבצים, פירוק לפונקציות, שמות ברורים
- 15% ממשק משתמש - תפריט ברור, הודעות משתמש, שימוש נעים
- 20% אלגוריתם חכם לרמת קושי שניה
- 25% בחינה בעל פה - יכולת להסביר את הלוגיקה - מה עשיתם, הבנה עמוקה של הקוד

סה"כ – 100%

- 10% בונוס – גרפיקה יפה, רמת קושי מתקדם
- הציון המקסימלי הינו 100

בהצלחה במטלה!

תזכרו: המטרה היא להבין ולתרגל, לא רק להגיש.