**עיבוד אילוצים – עבודה מס' 2**

**תיאור מימוש האלגוריתמים:**

התחלנו את פתרון התרגיל במימוש בעיות אילוצים כפי שנלמד בכיתה. ראשית, הגדרנו את המחלקה האבסטרקטית אשר ממדלת לנו בעיית אילוצים. המחלקה בעלת השדות הבאים:

- המסמל את מספר המשתנים בבעיה.

- המסמל את גדול דומיין הערכים של כל משתנה (ההנחה כי גודל הדומיין זהה לכל משתנה).

*– ווקטור המשמש להשמות למשתנים (בדומה ל-).*

*– ווקטור דו מימדי המייצג לכל משתנה את הדומיין ההתחלתי שלו.*

*- משתנה המסמל את העלות המקסימלית של הבעיה.*

*– מטריצה במימד אשר בכל תא שלה יש המבטאת את האילוצים בין משתנים (. עבור זוג ערכים מהדומיין של ישנו מספר חיובי בטווח - המבטא את משקל האילוץ (בעיות ). עבור בעיות של הטווח הוא בפרט .*

*– שדה בתוך הבעיה אשר סופר את מספר ההשמות אותן ביצענו בפתרון הבעיה.*

*– מונה הסופר את מספר הבדיקות אותן ביצענו במהלך הרצת האלגו'.*

*– משתנה השומר את ה הרנדומלי אותו אנו שולחים לבעיה, כדי ליצור את אותה בעיה שוב ושוב.*

*- משתנה בוליאני המסמל האם נמצא פתרון לבעיה או לא.*

*- כפי שהוגדר בתרגיל, המשתנה הנשמר הוא השכיחות/הסתברות של אילוץ בין משתנה למשתנה .*

*– כפי שהוגדר בתרגיל, המשתנה הנשמר הוא השכיחות של אילוץ בין ערך של משתנה לערך של משתנה .*

***–*** *משתנה השומר לנו את עלות הפתרון שנמצא לאחר הרצת האלגוריתם.*

*כתבנו את הפונקציה המאתחלת את מבני הנתונים הבסיסיים.*

*לאחר מכן מימשנו את*  המאתחלת את האילוצים על פי ההסתברויות המתאימות . על סמך היא קובעת האם יש צלע(אילוץ) בין שני משתנים. ולאחר מכן היא קוראת למתודה האבסטרקטית *אשר קובעת את מחיר הקשת בהתאם לבעיה.*

*עבור בעיות מחיר הקשת יהיה או 0 או 1. ועבור בעיות מחיר הקשת ינוע בטווח . כמובן שהוספנו .*

לאחר שמידלנו את הבעיות באופן כללי, יצרנו את בעיית ה- כתת-מחלקה של . השוני היחיד בה הוא שפונקציית אתחול האילוצים מאתחלת אותם על פי אילוצי בעיית N המלכות ולא לפי ההסתברויות הנתונות (משקל האילוץ הוא או 0 או 1,בהתאם).

המשכנו למימוש המחלקה כתת מחלקה של . דורסת את ,והאילוצים יאותחלו כפי שהגדרנו לעילא(*0 או 1)*.

וסיימנו עם מימוש המחלקה כתת מחלקה של . דורסת את ,והאילוצים יאותחלו כפי שהגדרנו לעיל().

השלב הבא היה לכתוב את מימושי האלגוריתמים לפתרון בעיות אילוצים.

לאחר קריאת המאמר של לה רוסה ומסגר החלטנו לממש את גירסת ה – אשר כתובה במאמר ().

יצרנו ממשק שנקרא ולו מתודה אחת . הגדרנו מחלקה הממשמת אותו - .כתבנו אותה עם חשיבה קדימה לקראת ולקראת ושילובם בקוד. לכן ישנם בקוד קריאות למתודות אשר פועלות כ – עבור ה  *הבסיסי, וידרסו בבנים (וכך למעשה תורכב עוד מדיניות על האלגו').*

התחלנו במימוש שלמעשה פשוט מכינה את הפרמטרים למציאת פתרון אופטימלי וקוראת לפונקציה .

נתבונן בפרמטרים הגלובליים המשמעותיים עבורנו לפתרון הבעיה:

–מאכסן את ההשמה הכי טובה שנמצאה.

- מכיל את משקל/מרחק הפתרון המינימלי.

*– זהו משתנה אשר מחזיק עבורנו בכל שלב את משקל הפתרון החלקי ,לא רלוונטי עבור .*

*האלגו' הקלאסי מבצע חיפוש לעומק של פתרון במשקל מינימלי ועובר על כל הבנים בעץ החיפוש ושומר את המסלול המינימלי אליו(דהיינו ההשמה). במימוש שלנו החלטנו לממש היורסיטיקה של כך שאם מצאנו פתרון בעלות 0 , אנחנו עוצרים את החיפוש וחוזרים למעלה.*

*הפונקציה אשר מוצאת את הפתרון ומעדכנת את הפרמטרים הגלובוליים המחזיקים אותו היא:*

מקבלת פתרון נוכחי, משקלו , המשתנה הבא לו נחפש השמה, ושאר המשתנים?

*למה אנחנו משתמשים ב ב ?*

*לאחר שהיה לנו את הגדרנו תת מחלקה בשם . כפי שהסברנו לעיל* כדי להרכיב את מדיניות ה - יצרנו שתי מתודות אצל האבא ,אשר כעת נדרוס אותן .ונחזיק כמובן מבנה נתונים אשר יסייע לנו בחישובים.

*– הוא ווקטור דו מימדי של*

קודם כל את החלקים המשותפים לכלל האלג' המתוארים במאמר, זאת מכיוון שבין האלגו' משתנות ונוספים מבני נתונים, אך גרעין האלגו' לא משתנה. הגדרנו את המחלקה האבסטרקטית – אשר מהווה אב קדמון לכלל האלג'. לה השדות:

*- מופע של הבעיה אותו אנחנו רוצים לפתור.*

*– המשתנה הבוליאני אותו אנחנו משנים תוך כדי ריצת האלגו',הבודק אם ההשמה קונסיסטינטית.*

*– המשתנה המסמל את שלבי פתרון הבעיה(יש פתרון/אין פתרון/לא ידוע).*

*- ווקטור דו מימדי המחזיק לכל משתנה את הדומיין הנוכחי שלו.*

*הגדרנו את המתודות האבסטרקטיות: ו- ואת המתודה הלא אבסטרקטית אשר משתמשת במתודות לעיל (מקבעת את הגוף הכללי של אלגו' חיפוש העצים של פרוסר, פרוצדורת ה-).*

*השלב הבא היה מימוש - מחלקות אשר יורשות מ - .*

*הינה מחלקה היורשת מ ,ופונקציות ה שלה הן הבסיסיות ביותר על פי המאמר.*

*אחרי שהיה לנו אלגוריתם בדוק ופותר בעיות אילוצים , המשכנו למימוש . המחלקה הנ"ל יורשת מ-.*

*ובה מבני הנתונים הבאים:*

*- שהוא ווקטור של (לנוחיות שלנו לקחנו קבוצה ממוינת כדי לשלוף את המקסימום בקלות מהמבנה נתונים). לכל הווקטור מכיל את ה של המשתנה ה-.*

*מימשנו את התוספות הנדרשות ל, מכיוון שכעת צריך לעדכן את מבני הנתונים של ה*  ולעדכן את הצורה בה אנחנו שולפים את המשתנה אליו אנו קופצים(, ).

לאחר מכן יצרנו את המחלקה אשר מממשת את האלגו' ( יורשת מ ).

נוספו המבני נתונים:

– כפי שהוגדר במאמר לכל משתנה מחסנית של קבוצות של *ערכים אשר הורדו מהדומיין הנוכחי שלו.*

*– ווקטור של מחסניות של integer. לכל משתנה ,שמורים המשתנים אשר הורידו ממנו ערכים ב במחסנית.*

*- ווקטור של מחסניות של integer. לכל משתנה ,שמורים המשתנים אשר הוריד מהם ערכים ב במחסנית.*

*הוספנו את הפונקציה .*

*ואת כל הפונקציות הדרושות לטיפול עם המבני נתונים ().*

*לאחר מכן, יצרנו את המחלקה*  אשר יורשת מהקודמת ובנוסף אליה מבצעת תוך פעולת ה- שלה. שוב דרסנו את המתודות הדרושות כדי להרחיב את הפעילות של האלגו'. הגענו למסקנה כי אין צורך להשתמש במבני נתונים נוספים. החלטנו כי אנו מטפלים בערכים שהורדו עם באותו אופן בו אנו מטפלים בערכים שהורדו ב . התבוננו בהגדרות של המבני נתונים וראינו כי שימוש במבני הנתונים הנוכחים יכולים לתת לנו את התפקוד הרצוי עבורנו בשביל שחזור הבעיה לאחר ביצוע לא מוצלח, ולכן אין צורך להוסיף עוד. טיפלנו בביטול השמה עם המתודה .

לצורך שמירת הסטטיסטיקות יצרנו את המחלקה אשר בה אנו משתמשים כדי ליצור ממוצעים של מספרי השמות/בדיקות בשביל 50 בעיות. עבור /.

לבסוף יצרנו את המחלקה אשר יוצרת הבעיות בצורה רנדומלית (עבור אותו נקבל את אותה בעיה).היא מריצה את האלגו' המתאימים על פי הדרישות (50 בעיות לכל שהוגדרו בתרגיל = סה"כ 1050 בעיות). וממלאת את הסטטיסטיקות.

**תיאור הניסויים:**

בשלב הראשון, ייצגנו את בעיית N-המלכות כבעיית אילוצים והרצנו עליהם את ארבעת האלגוריתמים שמימשנו כדי לבדוק את נכונות האלגוריתמים. חישבנו את מס' ההשמות ומס' הבדיקות של כל אחד מהאלגוריתמים על הבעיה הנ"ל עבור . התוצאות שהתקבלו (מס' ההשמות ומס' הבדיקות כתלות ב- N):

בשלב השני, עבור כל וכל יצרנו 50 בעיות והרצנו עליהם את FC-CBJ ו- FC-CBJ-DAC. חישבנו את ממוצע ההשמות וממוצע הבדיקות של כל אחד מהאלגוריתמים על כל קבוצה כזו של 50 בעיות. להלן התוצאות (מס' ההשמות ומס' הבדיקות כתלות ב- P2):

בנוסף, על מנת להשתכנע שהאלגוריתמים עובדים בצורה תקינה, בדקנו שכל בעיה נפתרת ע"י כל ארבעת האלגוריתמים או לא נפתרת ע"י כולם (על חלק מהבעיות לא הרצנו את אלגוריתם BT מכיוון שזמן הריצה של האלגוריתם עליהן היה ארוך מדי ולא ראינו בכך תועלת).

בדיקה נוספת שביצענו הייתה להפריד בין בעיות שנפתרו ע"י האלגוריתמים ובין בעיות שלא נפתרו על ידם. התוצאות שהתקבלו עבור למשל מראות על חלוקה ברורה לתחומים, כך שעבור ערכים נמוכים של P2 יש פתרונות לבעיות, עבור ערכים גבוהים של P2 אין פתרונות לבעיות וישנו תחום ערכים () שבו ישנה חפיפה, כלומר, ישנן בעיות שיש להן פתרון וכאלו שאין להן:

**מסקנות:**