ספר פרויקט –25.01.24

ממתין לאישור סופי של מיכל

בנושא:  
בניית מערכת שעות אופטימלית עם אילוצים

מגיש:  
דניאל סלמה ת.ז – 318266897

מנחה:  
 מיכל הורביץ

תוכן עניינים:

1. מבוא
2. הצגת הבעיה כתכנות ליניארי בשלמים
3. מבנה הפרויקט
4. טכנולוגיות וכלים
5. סיכום ומסקנות
6. רשימת נספחים
7. מבוא  
     
   מציאת פתרון אופטימלי עבור יצירת מערכת שעות בבית הספר הינה בעיה מורכבת וקשה. תהליך זה כרוך בהתחשבות באילוצים רבים. יצירה ידנית של לוחות זמנים מובילה לרוב לחוסר יעילות, התנגשויות תזמון וניצול לא מיטבי של משאבים. בעיה זו הופכת למאתגרת יותר ויותר ככל שהנתונים גדלים, הן מצד המערכת והן מצד האילוצים. משתנים מרובים חייבים להתאים בצורה מיטבית על מנת ליצור לוח זמנים ישים המתאים לצרכים של מורים, תלמידים וכיתות.

בפרויקט זה נבנה מודל אינטרנטי המבוסס על תכנון לינארי המסוגל להתמודד עם אילוצים שונים על מנת לפתור את הבעיה הבאה:

יש לשבץ  מורים והרצאות לפי דרישות המערכת עבור מספר שכבות וכיתות, תוך התחשבות באילוצים ודרישות שונות, ליצירת לוח זמנים יעיל ומאוזן הממקסם את ניצול המשאבים וממזער קונפליקטים.

מטרות הפרויקט:

* + 1. תזמון אוטומטי – מטרת הפרויקט היא להפוך את תהליך יצירת לוחות הזמנים של בית הספר לאוטומטיים, ולבטל את הצורך במאמצי תזמון ידני. זה לא רק חוסך זמן אלא גם מפחית את הסיכון לטעויות אנוש.
    2. יעילות – באמצעות אלגוריתמי אופטימיזציה מתמטיים, האפליקציה שואפת לייעל את ניצול המשאבים, כגון זמינות בכיתה ועומס העבודה של המורים. היא שואפת להקצות משאבים בצורה היעילה ביותר האפשרית.
    3. אפשרויות התאמה רחבה–האפליקציה נועדה להתאים לאילוצים והעדפות תזמון שונות, מה שהופך אותה למתאימה למגוון רחב של מוסדות חינוך, מבתי ספר יסודיים ועד אוניברסיטאות.
    4. ממשק ידידותי למשתמש – בכדי להפוך את תהליך התזמון לנגיש למנהלים ולמחנכים, הפרויקט מספק ממשק אינטרנט אינטואיטיבי שבו משתמשים יכולים להעלות נתונים, להציג מערכות שעות בצורה נוחה וקלה להבנה.

1. פתרון הבעיה באמצעות תכנון ליניארי

בעיית תכנון ליניארי היא בעיית אופטימיזציה של פונקציה ליניארית תחת אילוצים ליניאריים. כלומר, בהינתן משתנים ומערכת של אי-שוויונות בין משוואה ליניארית על הערכים לבין קבוע, למצוא את הערך המקסימלי או המינימלי שפונקציה ליניארית כלשהי יכולה לקבל תחת קיום של כל מערכת האי שוויונים.

תכנון ליניארי בשלמים (ILP - Integer Linear Programming) מציין בעיה דומה לבעיית התכנון הליניארי, אלא שהמשתנים חייבים לקבל ערכים שלמים דווקא. בניגוד לבעיית תכנון ליניארי רגילה שלה יש אלגוריתם פולינומי הפותר אותה , בעיה זאת היא NP קשה.

קיימים מספר אלגוריתמים אשר ניתן להשתמש בהם, בפרויקט זה בוצע עלי ידי הפותר של pulp המשתמש בפותר cbc , ובפותר simplex של ספריית gplk :

על מנת לפתור את בעיית המערכת השעות יש להגדיר מספר פרמטרים: משתני החלטה, פונקציית מטרה, אילוצים והגדרת הבעיה כלומר האם נרצה למקסם או לצמצם אותה.

הבעיה אותה נרצה לפתור היא תזמון מערכת שעות עבור בית ספר. בבעיה זו עלינו לתת פתרון למערכת האילוצים של מערכת השעות הנדרש, הכיתות, המורים, השכבות, הימים, משבצות הזמן והמקצועות. בתרגום לבעיית האופטימיזציה, פונקציית המטרה שלנו תהיה מיקסום מספר השיבוצים של מורים ומקצועות עבור כיתות ושכבות, תוך כדי מתן חשיבות עבור האילוצים השונים.

על סמך זאת נגדיר את הפרמטרים הנדרשים עבור פתרון בעיות ILP.

על סמך הגדרת הבעיה נרצה למקסם את תוצאות המערכת.

לכל מורה t, מקצוע s, יום d, שעהh , וכיתה g, מגדירים משתנה x, אשר יכול לקבל ערך 0 או ערך 1. המשתנה יקבל ערך 1 אם המורה t מלמד את הכיתה g את המקצוע s ביום d ובשעה h, אחרת יקבל 0.

מתמטית, בהינתן קבוצת המורים (), קבוצת המקצועות (), קבוצת הימים(), קבוצות הזמנים(), קבוצת הכיתות() וקבוצת השכבות().

נגדיר משתנה ההחלטה באופן הבא:

נגדיר משתנה x המייצג מורה המלמד מקצוע ביום ושעה ,כיתה השייכת לשכבה

x[teacher,lecture,time\_slot,day,classroom,grade]∈{0,1}

נקבל 1, עבור מורה teacher המלמד מקצוע lecture בשעה time ביום day בכיתה classroom ובשכבה grade העומד בכל האילוצים שיוגדרו בהמשך אחרת – 0, כך שתהליך האופטימיזציה אופציה לא תהיה חלק מהפתרון .

כעת, לאחר שהגדרנו את משתני ההחלטה ואת הגדרת הבעיה (מקסום / צמצום) נבנה את פונקציית המטרה שלנו, כלומר נרצה למקסם את ההשמות האפשריות במערכת השעות ולפיכך הפונקציה היא סכום כל משתני ההחלטה:

ולבסוף, נרצה להגדיר את האילוצים שעל פיהם יבחרו ההשמות האופטימליות

דוגמא לחלק מהאילוצים:

1. מורה לא יכול ללמד יותר מכיתה אחת באותה השעה ובאותו היום.
2. מורה לא יחרוג מכמות השעות שהוא המוקצות לו.
3. ניתן להגדיר ימי חופש למורה, בהם המורה לא מלמד.
4. מורה ילמד רק מקצועות מסוימים.
5. מספר השיעורים עבור כל מקצוע בכל כיתה יהיה על פי דרישות הנתונים.

ישנם אילוצים שקל להגדיר מתמטית (ופשוטים יותר לתכנות)

בכל זמן מורה יכול ללמד שיעור אחד בלבד

וישנם אילוצים מורכבים יותר למידול מתמטי כמו צמצום חלונות

באופן מעשי, ניתן לראות מימוש בpython באמצעות ספריית pulp

ספריית Pulp היא ספרייה בשפת python שמשמשת לבניית ופתרון בעיות אופטימיזציה. היא מסייעת בפתרון של מגוון רחב של בעיות מתמטיות ולוגיות, כולל בעיות ליניאריות, משתנים שלמים, ועוד.

עם Pulp, ניתן להגדיר משתנים, אובייקטיבים (פונקציית המטרה), ומגבלות ליניאריות ולוגיות. לאחר הגדרת הבעיה, כמו כן קיימים מגוון אלגוריתמי אופטימיזציה אשר ניתן להשתמש.

בקוד הבא אנו רואים את הגדרת הבעיה, הגדרת המשתנים, ופונקציית המטרה.

# Define the problem

prob = LpProblem("School\_Scheduling", LpMaximize)

# Define the decision variables

x = LpVariable.dicts("x", [(teacher, lecture, time\_slot,day, classroom, grade) for teacher in teachers for grade in grades for classroom in classrooms for lecture in lectures for time\_slot in time\_slots for day in days], cat='Binary')

# Define the objective function here

prob += lpSum(x)

בחלק זה ניתן לראות מספר אילוצים לדוגמה

for day in days:

for time in time\_slots:

for teacher in teachers:

# teacher can teach only one lecture at a time

prob += lpSum(x[(teacher, lecture, time, day, classroom, grade)] for grade in grades for classroom in classrooms for lecture in lectures) <= 1

for teacher in teachers:

if day in teachers[teacher][2]:

for time in time\_slots:

# teacher has days off

prob += lpSum(x[(teacher, lecture, time, day, classroom, grade)] for grade in grades for classroom in classrooms for lecture in lectures) == 0

for grade in grades:

for classroom in classrooms:

if classroom in grades[grade]:

for time in time\_slots:

prob += lpSum(x[(teacher, lecture, time, day, classroom, grade)] for teacher in teachers for lecture in lectures) <= 1

1. מבנה הפרויקט:

פרויקט זה בנוי משני חלקים backend ו- frontend, המקושרים על ידי Flask.

Flask היא ספרייה ליישומי אינטרנט שנכתבה ב- python. היא מספקת כלי פשוט וגמיש לפיתוח ובניית אפליקציות רשת.

ה fronted– נכתב ב- html ,css ו- js

בדף הראשי על המשתמש להעלות קובץ json ולאחריו יקבל 3 תצוגות של מערכת שעות. תצוגת מורים, כיתות/שכבות וימים.

שבהם המשתמש יכול לבחון את המערכת בהיבטים שונים.

דוגמא למבנה הקובץ:

Grades– מכיל בתוכו את דרישות המערכת, אילו שכבות וכיתות קיימים בו, בנוסף אילו מקצועות מלומדים ,כמה משבצות זמן עבור כל אחד מהם ומי המורה המלמד אותם.

Teachers– מערך המכיל את מספר השעות השבועיות המקסימליות עבור מורה, מקצועות שהוא מלמד, וימי חופש שבהם הוא לא ילמד.

Lectures– מערך של מקצועות

Classroom– מערך של כיתות

Time\_slots– מערך של זמנים

Days– מערך של ימים

דוגמא לקובץ נתונים:

{

"data": {

"grades": {

"A": {

"A1": {

"Science": [3,"Williams"],

"History": [3, "Johnson"],

"geography": [3, "Smith"],

"Math": [3, "Johnson"]

},

"A2": {

"Science": [3,"Williams"],

"History": [3, "Johnson"]

}

},

"B": {

"B1": {

"Science": [3,"Smith"],

"History": [3, "Johnson"],

"Math": [3, "Williams"]

},

"B2": {

"Science": [3,"Smith"],

"History": [3,"Smith"],

"Math": [3, "Williams"]

}

}

},

"teachers": {

"Smith": [16, ["Math", "History", "Science", "English", "Math"], [ "Friday", "Monday"]],

"Johnson": [16, ["English", "Math", "Science", " History"], ["Friday"]],

"Williams": [16, ["Science", "English", "geography", "Math"], [ "N"]]

},

"lectures": [ "History", "Science", "geography", "Math", "English"],

"classroom": ["A1","A2","B1","B2"],

"time\_slots": ["8-9", "9-10", "11-12", "12-13", "12-14", "14-15", "15-16", "16-17"],

"days": ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday"]

}

}

לאחר העלאת קובץ יתקבלו התצוגות הבאות:

תצוגת מורים, כלומר מערכת שעות המיועדת לכל מורה על מנת שידע בכל יום ושעה איזה כיתה הוא מלמד ואיזה מקצוע.

A school schedule with black text

Description automatically generated

תצוגה לפי ימים: מערכת זו מראה בכל יום ושעה איזה מורים מלמדים איזה מקצוע ואיזה כיתה.

A screenshot of a school schedule

Description automatically generated

תצוגה לפי שכבות וכיתות, תצוגת תלמידים– תצוגה המראה מערכת עבוד כל כיתה.

A school schedule with black text

Description automatically generated

הbackend נכתב בpython

תפקידו הוא לקרוא את הדאטה מקובץ הjson מבצע בדיקות מקדימות עבור הקלט על מנת לחסוך זמן ריצה, למשל שמספר שעות הלימוד שהמרצה קיבל לא חורג מהמקסימום הרשום לו.

במידה וקרתה שגיאה תוצג הודעת שגיאה.

אחרת, יתבצע אופטימיזציה על ידי cbc או simplex , במידה ו simplex ייכשל ינסה באמצעות cbc.

המערכת תארגן את הדאטה ותעביר זאת באמצעות flask לדפי הhtml המתאימים, ששם יבוצע תרגום הנתונים לידי בלאות ותצוגת מערכת השעות.

1. טכנולוגיות וכלים:
   * 1. Python 3– שפת התכנות המרכזית המשמשת את הפרויקט.
     2. Flask– מסגרת אינטרנט של python המשמשת ליצירת הקצה האחורי של האפליקציה.
     3. ספריית pulp– ספרייה לתכנות לינארי המשמשת לכתיבת האילוצים ולפתירתם.
     4. CBC solver – פותר ILP בקוד פתוח, משמש כברירת מחדל ב ספריית pulp
     5. GLPK solver – פותר חיצוני המשמש למשימות אופטימיזציה, חלק מספריית קוד פתוח המספק לפתרון בעיותLP ו- ILP משתמש בסימפלקס
     6. שימוש בשפות HTML, CSS ,JAVASCRIPT
2. **סיכום:**

פרויקט זה היה מאתגר מאוד. למדתי איך לקחת בעיה מחיי היום יום, לתרגם אותה למודל מתמטי ולפתור אותה באמצעות תכנון לינארי , כמו כן למדתי כיצד לעבוד עם שפות שונות ולממשק אותם יחד, חקרתי נושא חדש ללא ידע מקדים , עבדתי עם סביבות שונות ואיחדתי אותן על מנת שיעבדו יחדיו.

הקושי המרכזי בפרויקט הוא ביצוע ולידציה עבור הנתונים , מאחר ויש מספר אפשרויות של תשובות נכונות , כמו כן בעת הגדרת האילוצים יש לשים לב שהאילוצים אינם סותרים אחד את השני.

יש להגדיר באופן מדויק את הבעיה אותה רצינו לפתור ובכך ניתן להגדיר את פונקציית המטרה בצורה מיטבית.

**מסקנות:**

1. הפרויקט מיועד לסייע בפתרון בעיות רבות ומורכבות הנקשרות ליצירת לוחות זמנים בבתי הספר. המערכת יכולה להיות מותאמת למגוון רחב של מוסדות חינוך.
2. השימוש בטכנולוגיות מתקדמות כמו Flask, Pulp ו– GLPK Solver תורם לפרויקט ביצירת ממשק ידידותי ובפתרון אופטימיזציה מתמטית יעילה.
3. הפרויקט יכול להקל על תהליך תזמון השעות בבתי הספר, לחסוך זמן ומאמץ אנושי ולהפוך את זה למשימה פשוטה ומהירה.
4. ממשק המשתמש הידידותי מאפשר למנהלי בתי הספר ולמורים להיות ביציבה מעולה על מערכת השעות ולקבוע את הלוחות הזמנים בצורה יעילה.

סיכום זה משקף את היתרונות והתועלת הגדולה שבמערכת השעות האוטומטית שמתכוונת לספק. הפרויקט מביא לפתרון יעיל ומתמטי שמשפר את תהליך התזמון בבתי הספר ומציע פתרון מתקדם לבעיות הקשורות להקצאת משאבים בצורה אופטימלית.

1. רשימת נספחים:
2. <https://docs.pulpproject.org/pulpcore/>
3. <https://coin-or.github.io/pulp>
4. <https://flask.palletsprojects.com/en/3.0.x/>
5. <https://machinelearninggeek.com/solving-staff-scheduling-problem-using-linear-programming/>
6. <https://medium.com/walmartglobaltech/automating-shift-scheduling-with-linear-programming-fe1720f13620>
7. <https://towardsdatascience.com/schedule-optimisation-using-linear-programming-in-python-9b3e1bc241e1>
8. https://sites.laas.fr/files/MOGISA/2012-JourneesScientifiques/Artigues-MasterClass-Scheduling-LP-CPAIOR2012.pdf