

# מטלה 9 אלגוריתמים כלכליים:

## שאלה 4 תיעוד כללי:

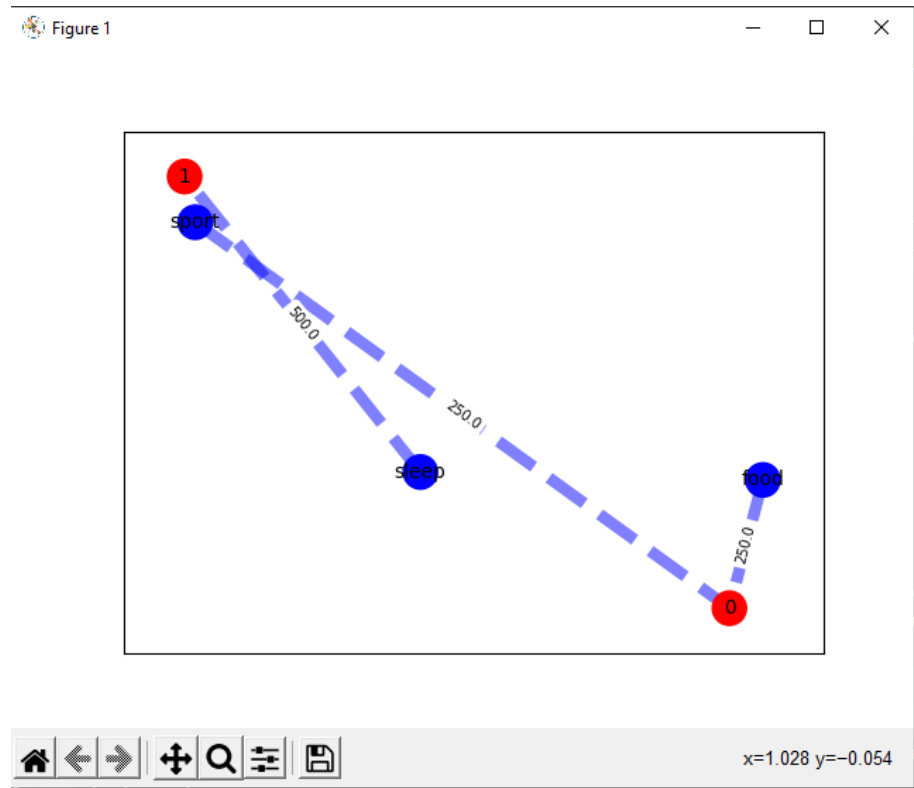
באופן כללי מימשתי בקוד את האלגוריתם האוטילטרי על תנאי לפי האלגוריתם חלוקת התקציב האישי של כל שחקן התבצעה לפי subject, מה subject שהוא ציין בהעדפות שלו, אשר היה לו את הביקוש הגבוה ביותר ( הביקוש נקבע לפי כמה שחקנים הציעו אותו subjects המועדפים עליהם, בנוסף כמובן שיכול להיות מצב שיש שיווין בביקוש של subjects, ויינתנו כמה subjects מועדפים).

הקוד מחזיר רשימה אשר מכילה את חלוקת התקציב הסופית לsubjects, לפי המיקום שלהם list של subjects שהפונקציה קיבלה.

ובנוסף מדפיס למסך את כמות התקציב שכל שחקן הקדיש לנושא מסוים ( יכול להיות מצב ששחקן ייתן תקציב לכמה נושאים אם היה שיווין ברמת הביקוש שלהם) כלי עזר להמחשת הקשרים באלגוריתם הוא שימוש בגרף וציורו במהלך הרצת הפונקציה. הגרף מכיל שני סוגי קודקודים. קודקודים אדומים אשר מייצגים את השחקנים וקודקודים כחולים אשר מייצגים את הנושאים.

הצלעות מייצגות את הבחירות של השחקנים בנושאים כנושאים מועדפים. משקל הצלע יהיה התקציב שנתן השחקן לאותו נושא. כלומר יהיה צלעות רק בין קודקודים של שחקן מסוים והנושאים שבחר כמועדפים

דוגמא:



```
total0 = 1000
subjects0 = ["food", "sport", "sleep"]
preferences0 = [{"food", "sport"}, {"sleep"}]
```

בדוגמא ניתן לראות כי ישנם שני שחקנים ( שחקן 0 ושחקן 1 )  
 ושלושה נושאים ( food, sleep, sport )  
 לשחקן 0 יש צלע עם food ועם sport שכל צלע בעלת משקל של 250  
 לשחקן 1 יש צלע עם sleep עם משקל של 500

## שאלה 4 תיעוד כללי לפונקציות:

הפונקציה הראשית הינה הפונקציה

Otilitery\_Budget(total: float, subjects: List[str], preferences: List[List[str]])

אשר מקבלת מספר כולל של סכום תקציב לחלק בין הנושאים.

רשימה של נושאים

ורשימה המציגת שחקנים אשר כל שחקן מחזיק רשימה של העדפות נושאים שלו  
 האלגוריתם תחילה עובר על כל העדפות וסופר את כמות הפעמים שכל נושא הוזכר בהעדפה  
 של שחקן מסוים .

לבסוף נקבל את כל הנושאים עם הכמות שהוזכרו.

לאחר מכן נעבור פעם נוספת על ההעדפות של השחקנים ונשמור לכל שחקן מי מהנושאים  
 המועדפים לו קיבל או קיבלו את הדירוג הכללי הגבוה ביותר.

לאחר מכן נחלק את התקציב האישי שלו לנושא שקיבל את הדירוג הכללי הגבוה ביותר.  
 ( במקרה שיש דירוג כללי זהה גבוה ביותר לכמה נושאים אזי התקציב האישי שלו יוחלק בין  
 נושאים אלו)

הפונקציה תדפיס את חלוקת התקציב לנושא של כל שחקן ובנוסף לבסוף תדפיס וגם תחזיר  
 רשימה מסודרת לפי מיקומים של התקציבים שקיבל כל נושא באופן כללי.

בנוסף בפונקציה זו יצויר גרף אשר מייצג את הקשרים בין השחקנים לנושאים לפי העדפות  
 השחקנים והתקציב שחולק

הפונקציה משתמשת בארבע פונקציות עזר: Show\_Graph, Init\_Graph, Get\_Subjects\_Score ,  
 Create\_Budget\_list

הפונקציה המשנית:

Get\_Subjects\_Score(preferences: List[List[str]], subjects\_score: Dict[str, int],  
 players\_num: int, graph: nx.Graph)

מילון של נושאים שתחילה הוא ריק והפונקציה תכניס אליו ערכים של כמות האיזכורים של נושא  
 בהעדפות של השחקנים.

ורשימה המציגת שחקנים אשר כל שחקן מחזיק רשימה של העדפות נושאים שלו  
 את מספר השחקנים

וגרף

הפונקציה עוברת על כל השחקנים ומעדכנת ושומרת את הכמות איזכורים של כל הנושאים לפי  
 העדפות של השחקנים.

בנוסף מוסיפה צלעות לגרף לפי הקישורים של שחקן לנושאים חשובים

פונקציה הקשורה באתחול בגרף:

Init\_Graph(graph: nx.Graph, subjects: List[str], number\_of\_players: int, color\_map:  
List[str])

רשימה של נושאים

מספר שחקנים

גרף

רשימה אשר תייצג את הצבע שכל קודקוד יקבל בציור הגרף  
הפונקציה מוסיפה את השחקנים כקודקודים עם מספר ואת הנושאים כקודקודים עם השם של  
הנושא, בנוסף מוסיפה צבעים לכל קודקוד .  
קודקודי השחקנים יהיו אדומים וקודקודי הנושאים כחולים

פונקציה הקשורה בהצגת ציור הגרף

Show\_Graph(graph: nx.Graph, color\_map: List[str])

גרף

רשימה אשר תייצג את הצבע שכל קודקוד יקבל בציור הגרף  
הפונקציה מיצרת רשימה של מקומות רנדומאליים  
ומגדירה את המיקומים, צבעים וכוונות של הקודקודים והצלעות ומיצרת את הגרף

פונקציה היוצרת את הרשימה הסופית המוחזרת מהפונקציה:

Create\_Budget\_list(subjects: List[str], subjects\_list\_budget: List[float],  
subjects\_full\_budget: Dict[str, int])

רשימה של נושאים

רשימה אשר יהיה בה את התקציב לכל נושא לפי מיקום הנושא ברשימה subjects בהתחלה  
רשימה זו ריקה

בנוסף מילון המכיל את התקציב שקיבל כל נושא לפי שם הנושא  
הפונקציה עוברת על כל הנושאים ומכניסה לרשימה לפי סדר הנושאים את התקציב אשר מופיע  
לו במילון