אלגוריתמים לחלוקה הוגנת של עוגות וקרקעות (1):

שיקול: הגינות.

הנחות: לכל המשתתפים ישנן זכויות שוות, אבל יש להם העדפות שונות.

חלוקה פרופורציונאלית – כל משתתף חושב שהחלק שלו שווה לפחות ½ מהשווי הכללי (יותק מ2 אנשים-1/n).

חלוקה ללא קנאה – כל משתתף חושב שהחלק שלו שווה לפחות כמו שאר החלקים (ללא קנאה גורר פרופורציונאלי).

אלגוריתם חתוך ובחר: (עבור שני ילדים, נניח עמי ותמי – פרופורציונאלי + ללא קנאה)

1. עמי חותך את העוגה לשני חלקים השווים בעיניו.
2. תמי בוחרת חלק אחד, ועמי מקבל את החלק שנשאר .

אלגוריתם "המפחית האחרון" (עבור n משתתפים – פרופורציונאלי, $O(n^2)$):

באלגוריתם זה, המשתתפים בוחרים סדר אקראי, אולי על ידי שליפת שמות מתוך כובע. לאחר מכן, השיטה מתקדמת באופן הבא:

1. האדם הראשון חותך פרוסה שהוא מעריך כחלק הוגן (פרוסה ששווה בעיניו בדיוק $1/n$).
2. האדם השני בוחן את החתיכה:
   א. אם הוא חושב שהחתיכה שווה פחות מחלק הוגן (שווה בעיניו לכל היותר $1/n$), אז הוא מעביר את החתיכה למשתתף הבא ללא שינוי.
   ב. אם הוא חושב שהחתיכה שווה יותר מחלק הוגן (החלק שווה יותר מ$1/n$), הוא חותך את החלק ומוסיף אותו בחזרה לערימה שיש לחלק, ומעביר למשתתף הבא את העוגה שנותרה (השווה בעיניו בדיוק $1/n$).
3. כל משתתף שנותר, בתורו, יכול להעביר את העוגה (שווה בעיניו לכל היותר $1/n$) או לקצץ אותה (שווה בעיניו לפחות $1/n$).
4. לאחר שהמשתתף האחרון קיבל את החלטתו, המשתתף האחרון שבחר לקצץ את החתיכה מקבל אותה. אם אף אחד לא קיצץ את הפרוסה, מי שחתך אותה (המשתתף הראשון) יקבל אותה.
5. מי שמקבל את החתיכה יוצא מהסבב עם החתיכה שלו והתהליך חוזר על עצמו עם שאר המשתתפים עד שיישארו רק 2 אנשים - הם יכולים לחלק את מה שנשאר ע״י האלגוריתם הקודם.

אלגוריתם אבן-פז (עבור n משתתפים – פרופורציונאלי, $O(n\, log_2 n)$):

1. אם נשאר רק שחקן אחד: הוא מקבל את כל העוגה .
2. אם יש יותר משחקן אחד, ומספר השחקנים זוגי:
   א. מבקשים מכל שחקן לסמן קו המחלק את העוגה לשני חצאים שווים בעיניו.
   ב. חותכים את העוגה בחציון של n הקווים.
   ג. שולחים כל שחקן לחצי שמכיל את הקו שלו - $\frac{n}{2}$ שחקנים לצד ימין ו-$\frac{n}{2}$ שחקנים לצד שמאל.
   ד. מחלקים כל חצי באופן רקורסיבי בין ה-$\frac{n}{2}$ שחקנים.
3. אם מספר השחקנים אי זוגי: מריצים צעד אחד של אלגוריתם "המפחית האחרון" וחוזרים לסעיף 2.

אלגוריתם סלפרידג'-קונווי (עבור 3 משתתפים – חלוקה ללא קנאה אבל החלוקה לא קשירה ועם שארית):

נניח שאנחנו מתחילים כמו באלגוריתם "חתוך ובחר":

מבקשים משחקן אחד (נניח, עמי) לחלק את העוגה לשלושה חלקים שווים בעיניו, ומבקשים משני השחקנים האחרים (תמי ורמי) לבחור את הפרוסה הטובה ביותר בעיניהם.

המקרה הקל הוא, שתמי ורמי בוחרים פרוסות שונות - אז כל אחד לוקח את הפרוסה שבחר, ועמי את הפרוסה השלישית, ואף אחד לא מקנא באחרים.

אבל מה אם רמי ותמי בוחרים את אותה פרוסה?

אז כנראה שהפרוסה הזאת היא גדולה מדי - צריך "לקצץ" אותה.

נבקש מתמי (נניח) לקצץ את הפרוסה כך שתהיה בעיניה שווה לפרוסה השנייה.

עכשיו אנחנו בטוחים שקיימת חלוקה ללא קנאה.

נחלק את השארית באופן הבא:

רמי מחלק את השארית לשלושה חלקים שווים בעיניו.

השחקנים בוחרים פרוסות לפי הסדר: תמי - עמי - רמי.

תמי לא מקנאת כי היא בחרה ראשונה, עמי לא מקנא בתמי (כיוון שהפרוסה של עמי שווה כמו פרוסה 1 לפני הקיצוץ, הרי שעכשיו עמי לא יקנא בתמי בשום מקרה, גם אם היא תקבל את כל השארית) וגם ברמי (כי הוא בחר לפניו), ורמי לא מקנא כי כל הפרוסות שוות בעיניו - מצאנו חלוקה ללא קנאה של כל העוגה!

אלגוריתם סימונס-סו (עבור n משתתפים – חלוקה כמעט ללא קנאה עם פרוסות קשירות):

1. נחלק את סימפלקס-החלוקות לסימפלקסונים קטנים, שאורך הצלע של כל אחד מהם הוא מילימטר. התהליך הזה נקרא מישלוש (triangulation).
   עכשיו יש לנו מספר סופי של נקודות (קודקודי הסימפלקסונים).
2. נשייך כל קודקוד לאחד השחקנים, כך שבכל סימפלקסון, כל השחקנים מיוצגים.
3. עבור כל קודקוד, נשאל את השחקן שהקודקוד שייך לו "איזו פרוסה אתה מעדיף – 1, 2 או 3?
   ונסמן את המספר על קודקוד.
4. קיבלנו תיווי (labeling) של המישלוש. נמצא סימפלקסון שבו כל התויות שונות, כל נקודה בסימפלקסון הזה מייצגת חלוקה ללא-קנאה עד-כדי-מילימטר, כמו שרצינו.

הלמה של ספרנר (Sperner's Lemma):

אם בכל קודקוד ראשי ישנה תוית אחרת, וגם בכל פאה - התויות זהות לתויות שבקודקודי הפאה,

אז קיים מספר אי זוגי של סימפלקסונים שבהם כל התויות שונות.

בפרט, קיים לפחות סימפלקסון אחד כזה.
נשים לב: סימפלקסון החלוקות מקיים את התנאי של ספרנר ולכן קיים סימפלקסון עם כל התויות, והוא מתאים לחלוקה ללא-קנאה-בקירוב.

האם קיים אלגוריתם סופי המוצא חלוקה ללא קנאה? התשובה היא לא!

סיכום:
אם עובדים עם שחקנים שהם "שמחים בחלקם" ורק רוצים לקבל את החלק הפרופורציונלי שלהם - זה יחסית פשוט - אפשר להשיג את זה בזמן $O(n\ log_2 n)$ועם פרוסות קשירות ע״י אלגוריתם אבן פז.
אבל, אם השחקנים קנאים וכל אחד מסתכל על "הדשא של השכן" - המצב הרבה יותר קשה - אין שום אלגוריתם סופי המבטיח לכולם פרוסות קשירות, וגם בלי דרישת הקשירות, האלגוריתמים מאד מסובכים ודורשים המון זמן.

יעילות כלכלית (2):
שיקול: יעילות
הנחות: (1)אין חשיבות לתכונת הקשירות, (2)מספיק להחליט כמה מכל משאב המשתתף מקבל (לא משנה איזה חלק בדיוק הוא מקבל), (3)המשאבים רציפים (ניתן לתת לכל שחקן אחוז כלשהו מכל משאב), (4)לכל שחקן ישנה פונקציית ערך רציפה (תוספת מעט משאב משפיעה רק במעט על הערך), (5)פונקציית הערך היא אדטיבית ומונוטונית עולה (אם עמי מייחס 100 לחפץ א ו50 לחפץ ב אזי הוא מייחס 150 לחפצים א+ב).

שיפור פארטו - מצב א׳ נקרא שיפור פארטו של מצב ב׳, אם מצב א טוב יותר לחלק מהמשתתפים, וטוב לפחות באותה מידה לכל השאר.
יעיל פארטו - מצב נקרא יעיל פארטו אם לא קיים מצב אחר שהוא שיפור פארטו שלו.

יעילות פארטו בחלוקת קרקעות ועוגות:
אלגוריתם הדיקטטורה (יעיל פארטו אם הדיקטטור מייחס ערך חיובי לכל משאב אך לא הוגן):
האלגוריתם בוחר את אחד מהשחקנים באופן שרירותי (כגון: לפי גיל, או באקראי), ונותן לו את כל המשאבים.
אם ישנם משאבים שהדיקטטור מייחס להם ערך אפס, אז אלגוריתם הדיקטטור אינו מחזיר חלוקה יעילה-פארטו, אבל אפשר בקלות לתקן אותו:
נותנים לדיקטטור את כל המשאבים שהוא מייחס להם ערך חיובי, ממנים דיקטטור חדש, ונותנים לו לבחור, מבין המשאבים שנשארו, את כל המשאבים שהוא מייחס להם ערך חיובי.
ממשיכים כך עד שכל המשאבים נלקחו. אלגוריתם זה נקרא דיקטטורה סדרתית.

יעילות אוטיליטרית - מצב א נקרא אוטיליטרי אם סכום הערכים של כל השחקנים במצב א גדול לפחות כמו סכום הערכים של כל השחקנים בכל מצב אחר (חלוקה שכזו תיתן כל משאב למשתתף שהערך שלו למשאב הוא הגדול ביותר).
הערה: הכלל האוטיליטרי הגיוני רק כאשר ישנה דרך אובייקטיבית למדוד את הערך של כל אחד מהמשתתפים, כך שכל הערכים נמדדים באותן יחידות. בנוסף כל מצב אוטיליטרי הוא יעיל פארטו.

אלגוריתם לחלוקה אוטיליטרית (חלוקה לא הוגנת אך יותר הוגנת מאלגוריתם הדיקטטור):
- תן כל אחד מהמשאבים לשחקן שעבורו הערך של האזור הזה הוא הכי גדול.

אלגוריתם לחלוקה אוטיליטרית במקרה הכללי: (כאשר פונקציות הערך של המשתתפים אינן אדטיביות)
נפתור בעיית אופטימיזציה קמורה בעזרת הספרייה cvxpy, הבעיה המתאימה לחלוקה אוטיליטרית של משאב כלשהו C בין n שחקנים היא:
$$Maximize \quad v_1(X_1)+\cdots+v_n(X_n) \qquad such\ that \quad (X_1,\dots,X_n) is\ a\ partition\ of\ C$$
מגדירים שלושה משתנים x, y, z המציינים איזה חלק מהמשאבים (עצים, דלק, ברזל בהתאמה) ניתן לעמי. החלק שניתן לתמי מכל משאב הוא $1-x, 1-y, 1-z$ בהתאמה.
מחשבים את התועלת של עמי ותמי במשתני-עזר. מגדירים בעיית מקסימיזציה של סכום התועלות. מגדירים את האילוצים, והם, שערכי כל המשתנים הם בין 0 ל-1.

יעילות אגליטרית - מצב א נקרא אגליטרי אם הערך המינימלי של שחקן במצב א גדול לפחות כמו הערך המינימלי של שחקן כלשהו בכל מצב אחר.
הערה: אם פונקציית-הערך של כל שחקן היא רציפה ומונוטונית-עולה-ממש, אז בכל חלוקה אגליטרית, כל השחקנים מקבלים את אותו ערך בדיוק.
חלוקה אגליטרית, היא הוגנת אך אינה בהכרח יעילה פארטו.

לקסימין-אגליטרי - מצב א נקרא לקסימין-אגליטרי אם הוא ממקסם את הערך הקטן ביותר, בכפוף לזה, הוא ממקסם את הערך השני הכי קטן, בכפוף לזה, הוא ממקסם את הערך השלישי הכי קטן, וכן הלאה.
הערה: כל מצב לקסימין-אגליטרי הוא יעיל-פארטו (וגם פרופורציונאלי) אך אינו ללא קנאה.

אלגוריתם לחישוב חלוקה אגליטרית:
על-פי הגדרה, בעיית אופטימיזציה המוצאת חלוקה אגליטרית היא:
נוסיף משתנה נוסף z, המייצג את הערך המינימלי, נוסיף אילוצים הקובעים ש- z הוא אכן הערך המינימלי, כלומר, ש- z קטן מכל n ערכי השחקנים ואז למקסם את z:

$$Maximize \quad z$$
$$such\ that \quad (X_1,\dots,X_n) is\ a\ partition\ of\ C\ \&\ z\le v_1(X_1),\dots,z\le v_n(X_n)$$

חישוב חלוקה לקסימין-אגליטרית/אלגוריתם וילסון (3 משאבים ו4 משתתפים):

- מחשבים חלוקה אגליטרית, נניח שהמקסימום = z (מכאן: בחלוקת לקסימין, יש שחקן אחד לפחות שמקבל בדיוק z, וכל השאר מקבלים לפחות z).
- עבור כל שחקן נחשב את הערך המקסימלי שהוא יכול לקבל, תחת האילוץ שכל שאר השחקנים מקבלים לפחות z.
- אם הערך המקסימלי המתקבל עבור שחקן מסויים הוא z, אז השחקן "רווי" - הערך שלו בחלוקת לקסימין = בדיוק z.
- נחשב חלוקה אגליטרית עבור כל השחקנים שנשארו לא רווים, תחת האילוץ שהערך של כל השחקנים הרווים הוא בדיוק ערך הרוויה שלהם.
- נמשיך כך עד שכל השחקנים הופכים להיות רווים.

נריץ את האלגוריתם על הדוגמה:

- **שלב 1:**
  ערך אגליטרי = 3.
  ערכי מקסימום לשחקנים - א, ב, ג, ד = 4, 3, 8.25, 8.25.
  שחקו ב רווי.
- **שלב 2 (נשארו שחקנים א, ג, ד):**
  ערך אגליטרי = 4.
  ערכי מקסימום לשחקנים - א, ג, ד = 4, 6, 6.
  שחקן א רווי.
- **שלב 3 (נשארו שחקנים ג, ד):**
  ערך אגליטרי = 5.
  ערכי מקסימום לשחקנים - ג, ד = 5, 5.
  כולם רווים. -סיימנו!

משאב:	**ברזל**	**דלק**	**עצים**
א:	4	0	0
ב:	0	3	0
ג:	5	5	10
ד:	5	5	10

פסאודו-קוד של האלגוריתם הכללי לחלוקה לקסימין אגליטרית:

1. אתחול: S =: קבוצה ריקה (קבוצת השחקנים הרוויים).
2. פתור את בעיית האופטימיזציה הבאה:

$$\begin{aligned}&Maximize && z\\ &such\ that && (X_1,\dots,X_n)\ is\ a\ partition\ of\ C\\ &&& v_i(X_i) = saturated-value[i]\ \ for\ all\ i\in S\\ &&& z\le v_j(X_j)\ for\ all\ j\notin S\end{aligned}$$

יהי z_{max} הערך המקסימלי שהתקבל בבעיה זו.

3. לכל שחקן חופשי j, פתור את בעיית האופטימיזציה הבאה:

$$\begin{aligned}&Maximize && z\\ &such\ that && (X_1,\dots,X_n)\ is\ a\ partition\ of\ C\\ &&& v_i(X_i) = saturated-value[i]\ \ for\ all\ i\in S\\ &&& z_{max}\le v_j(X_j)\ for\ all\ j\notin S\end{aligned}$$

4. אם הערך המתקבל שווה ל- z_{max}, אז jהוא שחקן רווי: הוסף אותו ל-S ושמור את ערך-הרוויה שלו
   אם כל השחקנים רוויים – סיים והחזר את החלוקה $(X_1,\dots,X_n)$ אחרת – חזור לשורה 2.

חלוקה הוגנת של חפצים בדידים (3):

שיקול: חלוקה כמעט הוגנת (כשיש 101 חפצים זהים, ניתן 50 לשחקן אחד ו-51 לשחקן השני, כך שהחלוקה תהיה ללא קנאה עד-כדי חפץ אחד).
הנחות: חפצים בדידים (לא ניתן לחתוך, כגון תכשיטים משימות לביצוע, חלוקת חדרים ועד) ושונים כאשר המטרה לשתף כמה שפחות חפצים – ניתן פיצוי כספי למי שמקבל פחות חפצים.

מודל אורדינלי – הנחות: כל אחד מהשותפים יודע להגיד, עבור כל וקטור מחירים, איזה חדר הוא מעדיף בוקטור המחירים הזה, החדרים סבירים (כל דייר מוכן לקבל לפחות חדר אחד), השותפים עניים (מעדיפים חדר בחינם ולא בתשלום). חסרון: הנחת הדיירים העניים.
הגדרת חלוקה ללא קנאה לכל i,j: $V_i(X_i)-P(X_i)\ge V_i(X_j)-P(X_j)$

אלגוריתם לחלוקת חדרים – (2 משתתפים, 2 חדרים, ללא קנאה)
שותף אחד מחליט מה יהיה המחיר של כל חדר, והשותף השני בוחר חדר.

אלגוריתם לחלוקת חדרים – סימונס סו (n משתתפים, n חדרים, ללא קנאה אך עובד רק במודל האורדינלי)
נניח ששכר-הדירה הכולל הוא R.
כל חלוקה של שכר-דירה בין החדרים היא וקטור של מחירים שסכומם R.
ניקח את סימפלקס היחידה, ונתאים לכל נקודה בסימפלקס $(x_1,\dots,x_n)$, וקטור של מחירים $(p_1,\dots,p_n)$ כאשר לכל i: $p_i = R*x_i$
שימו לב שבכל נקודה בסימפלקס, סכום המחירים הוא בדיוק R, ולכן כל נקודה מתאימה לחלוקה אפשרית של שכר-הדירה בין החדרים.
עכשיו נבצע מישלוש של הסימפלקס - נחלק אותו לסימפלקסונים קטנים, נניח, בגודל של אגורה אחת. עבור כל וקטור-מחיר שנמצא על קודקוד של המישלוש, נשאל כל אחד מהשותפים "איזה חדר אתה מעדיף?".
הנחת "הדיירים העניים" אומרת, שבכל קודקוד ראשי, כל דייר יבחר את אחת הפרוסות הריקות (= אחד מ n-1 החדרים שהמחיר שלו בקודקוד זה הוא 0).
לדוגמה, אם יש שלושה חדרים ושלושה דיירים, אז בקודקוד מספר 1, כל דייר יכתוב 2 או 3. באותו אופן, בקודקוד 2 כל דייר יכתוב 1 או 3, ובקודקוד 3 כל דייר יכתוב 1 או 2.

בקוים המחברים בין הקודקודים, כל דייר יבחר את החדר שהמחיר שלו 0.
למשל, בקו בין קודקוד 1 לקודקוד 2, כל דייר יבחר את חדר מספר 3, וכו' (זה בדיוק הפוך מהמצב בבעיית חלוקת העוגה).
אנחנו רוצים להשתמש בלמה של ספרנר, אבל לשם כך אנחנו צריכים להחליט איזו תוית תהיה על כל קודקוד ראשי.
אם נבחר בכל קודקוד מספר אחר, התויות על הצלעות יתאימו לתנאי של ספרנר.
אפשר לבחור תויות באופן דומה גם כשיש n חדרים ו-n שותפים.
לכן, לפי הלמה של ספרנר, קיים סימפלקסון שבו כל דייר בוחר חדר אחר. הסימפלקסון הזה מייצג חלוקה כמעט-ללא-קנאה.

מודל קרדינלי - כל אחד מהשותפים מייחס ערך מספרי לכל חדר, המשקף את הסכום המרבי (לחודש) שהוא מוכן לשלם עבור החדר, החדרים סבירים, הדיירים קוואזי לינארים (אם דייר חושב שחדר מסויים שווה v, ושכר-הדירה על החדר הזה הוא p, אז התועלת של הדייר היא v פחות p).
חסרון
חלוקה ללא קנאה היא חלוקה שבה כל שותף מקבל חדר עם תג-מחיר, כך שהתועלת של כל שותף מהחדר שלו גדולה לפחות כמו התועלת שלו בכל חדר אחר.

משפט: בכל השמה ללא קנאה, סכום הערכים של הדיירים בחדרים שהם גרים בהם הוא מקסימלי והחלוקה תישאר ללא-קנאה לכל השמה ממקסמת-סכום-ערכים. לכן כדי למצוא חלוקת שכר דירה ללא קנאה, הכרחי ומספיק למצוא השמה הממקסמת את סכום הערכים.

אלגוריתם סונג-ולאך: (חסרון – יכול להחזיר תשלום שלילי והדיירים בוודאי לא יסכימו לשלם למישהו כדי שיגור בדירה - הם יעדיפו לוותר על החדר)
1. נמצא השמה של אנשים לחדרים, הממקסמת את סכום הערכים: עבור n קטן נעבור על כל $n!$ האפשרויות, עבור n גדול נפתור בעיית שידוך מושלם עם משקל מקסימלי – קלט: גרף דו״צ צד אחד הדיירים ובצד השני החדרים וצלעות ממושקלות בן דייר לחדר עם הערך שהוא מייחס לו
2. נקבע מחיר לכל חדר, כך שההשמה עם המחירים תהיה ללא קנאה: קביעת המחירים תהיה על ידי פתירת הבעיה הבאה (ניתן לפתור עם *cvxpy* $d[i]$ מציין את הדייר המשויך לחדר מספר i (לפי פתרון בעיית השידוך עם משקל מקסימלי):

$Minimize\ 1 \quad such\ that \quad For\ all\ i,j\colon w[d[i],i]-p[i] > w[d[i],j]-p[j] \quad \&\&\ \ sum_i\ p[i] = TotalRent$

פתרון בעיית הטרמפיסט: (מה לעשות כשחוזר רך תשלום שלילי מאלגוריתם סונג-ולאך?)
להפוך את כל המחירים השליליים ל-0, ולחלק את היתרה בין השחקנים האחרים.
הבעיה היא, שהפתרון יהיה עם קנאה. אבל, הדיירים היחידים שיקנאו יהיו הדיירים שמקבלים חדר בחינם.
האם אפשר למצוא פתרון שהוא גם ללא-קנאה וגם בלי טרמפיסטים?
התשובה היא כן - הפתרון האורדינלי. אבל לפתרון האורדינלי יש בעיה אחרת - הוא מניח את הנחת "הדיירים העניים".

חלוקה הוגנת בקירוב (4):
שיקול: חלוקת חפצים בדידים – בלתי אפשר להבטיח חלוקה הוגנת לחלוטין אך נשאף למקסימום הגינות.

אלגוריתם לחלוקת מושבים בכנסת (2 מפלגות בלבד, זכויות שונות וחפצים זהים):
לעגל את מספר המושבים של כל מפלגה לשלם הגדול ביותר.

שיטת השארית הגדולה ביותר/שיטת המילטון(n מפלגות, חסרון: אינה עקבית, זכויות שונות וחפצים זהים):
1. נותנים לכל מפלגה את מספר המושבים השלם המגיע לה (המספר מעוגל כלפי מטה)
2. מחלקים את המושבים העודפים למפלגות לפי סדר יורד של השארית.

עקביות - שיטה לחלוקת-מושבים נקראת עקבית, אם עבור כל תת-קבוצה X של מפלגות, שקיבלו ביחד n מושבים בחלוקה הכללית – אם נשתמש באותה שיטה כדי לחלק את n המושבים בין המפלגות בקבוצה X בלבד, נקבל אותה חלוקה בדיוק כמו בחלוקה הכללית.

שיטת ג׳פרסון (n מפלגות, עקבית, זכויות שונות וחפצים זהים):
1. אתחול: כל מפלגה מקבלת 0 מושבים.
2. כל עוד נשארים מושבים פנויים: מחשבים, עבור כל מפלגה, את המנה (מספר קולות) / (מספר מושבים נוכחי + 1). נותנים את המושב הבא למפלגה שעבורה המנה הזאת גדולה ביותר.

שיטת מחלק – הכללה לג׳פרסון (n מפלגות, עקבית לכל פונקציה f, זכויות שונות וחפצים זהים):
בוחרים פונקציה כלשהי f, המייחסת לכל מספר שלם s, מספר ממשי כלשהו בין s לבין $s+1$.
המושבים מחולקים באופן הבא:
1. אתחול: כל מפלגה מקבלת 0 מושבים
2. כל עוד נשארים מושבים פנויים: מחשבים, עבור כל מפלגה, את המנה $\frac{\text{מספר הקולות}}{f(s)}$, כאשר s הוא מספר המושבים הנוכחי של המפלגה. נותנים את המושב הבא למפלגה שעבורה המנה הזאת גדולה ביותר.

אדאמס(טובה לקטנות): $f(s)=s$, הנטיגנטון-היל: $f(s)=\sqrt{s*(s+1)}$, וובסטר(הוגנת): $f(s)=s+\frac{1}{2}$, ג׳פרסון(טובה לגדולות): $f(s)=s+1$.

חלוקה הוגנת בקירוב מלבד חפץ אחד:
EF1 - חלוקה נקראת “ללא קנאה מלבד 1” אם לכל שני שחקנים i,j קיים חפץ בסל של שחקן j, שאם נסיר אותו - אז שחקן i לא יקנא.
אלגוריתם הסבב (לחפצים בדידים כאשר הזכויות שוות והחפצים שונים - מחזיר חלוקה EF1):
1. מסדרים את השחקנים בסדר שרירותי כלשהו.
2. כל שחקן לוקח, מבין החפצים שנשארו, את החפץ שהוא הכי רוצה.
3. אם נשארים חפצים – חוזרים לשלב 2.

WEF - חלוקה נקראת "ללא קנאה משוקללת", אם לכל שני שחקנים i,j עם זכויות w_i, w_j, מתקיים: $\frac{v_i(X_i)}{w_i} \geq \frac{v_i(X_j)}{w_j}$
כלומר: שחקן i מעריך את הסל שלו, ביחס למשקל שלו, לפחות כמו שהוא מעריך את הסל של כל שחקן אחר j, ביחס למשקל של j.

אלגוריתם לחלוקת חפצים שונים כאשר הזכויות שונות (מחזיר חלוקה ללא קנאה משוקללת עד כדי חפץ אחד):
כל עוד יש חפצים פנויים:

1. מחשבים, עבור כל שחקן, את המנה $\frac{\text{הזכות}}{f(s)}$, כאשר s הוא מספר החפצים הנוכחי של השחקן.
2. השחקן, שעבורו המנה הזאת גדולה ביותר, לוקח מבין החפצים שנשארו את החפץ שהוא הכי רוצה.

חלוקה הוגנת עם שיתוף מינימלי (5):
הנחה: נרצה להשאיר (כמות מינימלית של) חפצים בבעלות משותפת כאשר אי אפשר לחלק אותם

אלגוריתמים לחלוקה הוגנת עם שיתוף חפצים מינימלי (2 שחקנים):

- אלגוריתם 1 (ללא קנאה, חפץ אחד לכל היותר נחתך, לא יעילה פארטו בכל המקרים):
  נסדר את כל החפצים בשורה ונתייחס אליהם כמו עוגה. נבקש מאדם אחד לחתוך ומהשני לבחור.
- אלגוריתם 2 (יעילה פארטו, ממקסמת סכום ערכים וללא שיתופים אך עלולה להיות קנאה):
  כל חפץ נמסר למי שמייחס לו את הניקוד הגבוה ביותר.
- אלגוריתם 3 (יעילה פארטו, ללא קנאה אך יכול להיות שיחתך יותר מחפץ אחד):
  נתייחס לחפצים כמו לסחורות, ונמצא חלוקה הממקסמת את מכפלת הערכים.

אלגוריתם המנצח המתוקן (מוצא חלוקה הוגנת, שוויונית (סכום הנקודות של כל שחקן יהיה שווה), יעילה פארטו, ללא קנאה ועם שיתוף חפץ אחד לכל היותר, m חפצים, 2 אנשים המייחסים ערך שונה לכל חפץ):

1. עבור כל חפץ, חשב את יחס הניקוד בין דונאלד לאיואנה. סדר את החפצים מימין לשמאל בסדר עולה של יחס זה - כך שבצד ימין נמצאים החפצים שאיואנה מייחסת להם ניקוד גבוה יותר, ובצד שמאל - החפצים שדונאלד מייחס להם ערך גבוה יותר.
2. אתחול: תן את כל החפצים לדונאלד.
3. עבור על החפצים מימין לשמאל. העבר חפץ אחר חפץ לאיואנה. חשב את סכום הנקודות שאיואנה מייחסת לחפצים שברשותה, ואת סכום הנקודות שדונאלד מייחס לחפצים שברשותו. אם הסכומים של שני השחקנים שווים - סיים.
4. אם הגעת לחפץ, שאם יתנו אותו לאיואנה - סכום הנקודות שלה יהיה גדול יותר, ואם יתנו אותו לדונאלד - סכום הנקודות שלו יהיה גדול יותר, חלק אותו ביחס שיגרום לסכום הנקודות להיות שווה (פתרון משוואה בנעלם אחד).

גנריות – פונקציות ערך של שני שחקנים כלשהם נקראות גנריות אם כל יחסי הערכים שהם מייחסים ל-m החפצים שונים זה מזה.

אלגוריתם למציאת חלוקה עם שיתוף מינימלי בין 2 שחקנים עם פונקציות ערך גנריות (מוצא חלוקה יעילה וללא קנאה עם שיתוף אחד לכל היותר):

1. סדר את החפצים לפי סדר עולה של יחס הערכים.
2. העבר חפצים בהתאם לאלגוריתם "המנצח המתוקן".
3. אם תוך-כדי ההעברה התגלתה חלוקה ללא-קנאה בלי שיתוף כלל – החזר אותה.
4. אחרת, החזר חלוקה ללא-קנאה עם שיתוף אחד, כמו באלגוריתם "המנצח המתוקן".

משפט: נתונה חלוקת-משאבים כלשהי (נקרא לה חלוקה א), שבה ערכו של כל שחקן i הוא t_i.
קיימת חלוקת-משאבים אחרת (נקרא לה חלוקה ב) המקיימת את התנאים הבאים:

- הערך של כל שחקן i בחלוקה ב הוא לפחות t_i.
- בחלוקה ב יש לכל היותר $n-1$ חפצים משותפים.
- קיים אלגוריתם מהיר המוצא חלוקה ב המקיימת תנאים אלו.

הערה: בכל בעיית חלוקת משאבים עם n אנשים ניתן למצוא חלוקה פרופורציונאלית שבה לכל היותר $n-1$ מהמשאבים משותפים, ניתן לכל אחד 1/n מכל חפץ ונפעיל על החלוקה את המשפט הקודם וכך כל שחקן יקבל לפחות כמו שקיבל במקרה הקודם.
הערה: בכל בעיית חלוקת משאבים עם n אנשים עם זכויות שונות ניתן למצוא חלוקה פרופורציונאלית שבה לכל היותר $n-1$ מהמשאבים משותפים, נניח שהזכות של כל שחקן i היא w_i, וננרמל את הזכויות כך שסכומן יהיה 1. ניתן לכל שחקן i חלק w_i מכל משאב. נגדיר חלוקה זו כ"חלוקה א" ונפעיל עליה את המשפט הקודם.

אלגוריתמים מגלי-אמת/מכרזים (6):
שיקול: הערך של כל משתתף הוא מידע פרטי ונרצה אלגוריתמים שמעודדים את המשתתפים להגיד את הערך האמיתי שלהם.
הנחות: חפץ אחד, וכל שחקן j מייחס לו ערך אחר $v[j]$, אם שחקן זוכה בחפץ התועלת שלו היא הערך פחות כמה ששילם (קוואזי לינארי).

מגלה אמת - אלגוריתם נקרא מגלה-אמת אם לכל משתתף באלגוריתם כדאי תמיד להגיד את הערכים המגלה-אמתים שלו, בלי קשר לשאלה מה עושים האחרים.

סוגי מכרזים:
מכרז מחיר ראשון (לא מגלה אמת) - כל משתתף כותב מספר במעטפה ומגיש לכרוז. הכרוז פותח את כל המעטפות. מי שכתב את המחיר הגבוה ביותר זוכה, ומשלם את המחיר שהכריז.
מכרז שבו אף אחד לא זוכה אף פעם (מגלה-אמת, לא יעיל פארטו).
מכרז מחיר שני/ויקרי (מגלה אמת ויעיל פארטו) - השחקן שהכריז את הערך הגבוה ביותר זוכה, והוא משלם את המחיר השני בגובהו.

הנחות: m חפצים שונים (כמו פרסומות באינטרנט, פרסומת k בראש הדף שווה יותר – שיעור ההקלקה r_k שלה גבוה יותר) ולכל משתתף יש ערך הקלקה v_j – כמה הוא חושב שהוא ירוויח, בממוצע, מהקלקה על המודעה שלו.

אלגוריתם GSP (יעיל פארטו ולא מגלה אמת כשיש 2 או יותר חפצים, התשלומים גבוהים-טוב למפרסם):

1. נמצא הקצאה הממקסמת סכום ערכים – ע״י אלגוריתם חמדני: סדר את המפרסמים בסדר יורד של v_j, תן למפרסם הj את המקום הj.
2. נקבע את התשלומים של כל מפרסם – ע״י מכרז מחיר שני מוכלל (אינו מגלה אמת כשיש 2 מקומות או יותר): מפרסם שההכרזה שלו היא הj בגובהה, זוכה במקום j, ומשלם את ההכרזה של המפרסם ה$j+1$.

אלגוריתם VCG (יעיל פארטו ומגלה אמת תמיד, התשלומים נמוכים-פחות טוב למפרסם):

- בחר את התוצאה עם סכום-הערכים הגבוה ביותר.
- עבור כל שחקן: חשב את סכום הערכים של שאר השחקנים וחשב את סכום הערכים של שאר השחקנים אילו השחקן הנוכחי לא היה משתתף.
- גבה מהשחקן את ההפרש בין שני הסכומים.

אלגוריתם VCG למציאת מסלול זול ביותר (העברת חבילה ברשת תקשורת ולכל קשת יש עלות להעברת חבילה דרכה כאשר העלות היא פרטית):
אוסף התוצאות האפשריות הוא אוסף כל המסלולים מהמקור ליעד.

- השלב הראשון במכרז וק"ג הוא חישוב התוצאה עם סכום-הערכים הגדול ביותר.
  במקרה זה אנחנו מחפשים את סכום העלויות הקטן ביותר, אבל ההבדל הוא טכני בלבד - אפשר להציג כל עלות כערך עם סימן מינוס.
  לכן, אנחנו יכולים לבצע את השלב הראשון במכרז בעזרת אלגוריתם בלמן-פורד.
- השלב השני הוא חישוב התשלומים. כדי לחשב את התשלום של שחקן מסויים (במקרה זה, של קשת מסוימת), אנחנו צריכים לחשב מה היתה התוצאה אילו השחקן הזה לא היה משתתף. אנחנו צריכים למחוק (באופן זמני) את הקשת מהגרף, ולמצוא את המסלול הזול ביותר בגרף שהתקבל, כלומר להריץ שוב את אלגוריתם דייקסטרה. אחרי שמצאנו את המסלול הזה, אנחנו צריכים לחשב את סכום הערכים של כל שאר הקשתות, לחסר את סכום הערכים של כל שאר הקשתות בלי המחיקה, וההפרש הוא התשלום שתצטרך הקשת לשלם.

אלגוריתם VCG למילוי תרמיל (עובד רק כאשר מספר המפרסמים הוא קטן):
קודם-כל מחשבים את ההשמה הממקסמת את סכום הערכים.
ואז מחשבים – עבור כל חפץ/מפרסם – את ההשמה הממקסמת את סכום הערכים בלעדיו.

עוד אלגוריתמים מגלי-אמת (7):
הבעיה: בעיית התרמיל – חפצים עם ערכים שונים ומשקלים שונים, נרצה להכניס לתרמיל (עם קיבולת כלשהי) חפצים שערכם גבוה ביותר.

אלגוריתם (אם מספר החפצים קטן והערך של כל חפץ הוא ציבורי):
בדוק את כל תת-הקבוצות של חפצים, ולבחור את תת-הקבוצה עם הערך הגבוה ביותר מבין תת-הקבוצות שסכום המשקלים שלהן הוא לכל היותר 100.

שיקול: מקסום רווחים אך עבור מספר חפצים גדול זו בעיה NP-קשה ולכן נחפש אלגוריתמי קירוב.
הנחות: ערך החפץ ידוע רק לבעליו (רדיו, נקציב 100 שניות לפרסומות ונרצה לנצל את כל 100 השניות ולהרוויח הכי הרבה מהמפרסמים).
כיוון שהערכים הם מידע פרטי נרצה לבצע מכרז, השלב הראשון של VCG הוא למצוא את התוצאה עם סכום הערכים הגבוה ביותר, נראה כמה אפשרויות למציאת סכום ערכים מקסימלי בקירוב.

אלגוריתם חמדני א (עובד יפה כאשר כל החפצים באותו משקל, כשיש הבדלים הוא לא טוב ובמקרים מסוימים התשלומים גבוהים. לא מגלה אמת):
סדר את החפצים בסדר יורד של הערך.
בחר חפצים לפי הסדר עד שהתרמיל מתמלא.

אלגוריתם חמדני ב (עובד יפה במקרים שבהם א כושל אבל לפעמים מחזיר תוצאות לא טובות):
סדר את החפצים בסדר יורד של היחס ערך/משקל.
בחר חפצים לפי הסדר עד שהתרמיל מתמלא.

אלגוריתם א+ב (נותן קירוב ½ לבעיית התרמיל אך אינו מגלה אמת):
הפעל את שני האלגוריתמים החמדניים, ובחר את התוצאה עם הסכום הגבוה.

חיסרון גדול של מנגנון VCG - הוא חייב למצוא את סכום-הערכים המקסימלי, וחייב למצוא אותו במדוייק - לא בקירוב. אם מציאת המקסימום היא בעיה חישובית קשה - אנחנו לא יכולים להשתמש ב-VCG כי אנחנו לא מצליחים לחשב את השלב הראשון של האלגוריתם. צריך מכרז מסוג אחר.

מכרז מאירסון (מיועד למצבים שבהם לא יכולים/רוצים למקסם סכום ערכים, מגלה אמת אם כלל התשלום טובים):
קלט: כלל בחירה בינארי מונוטוני - כלל שמקבל את ערכי השחקנים לכל החפצים וקובע, לגבי כל שחקן, אם הוא "נבחר" או לא (1-נבחר, 0-לא).
פלט: כלל תשלום – כלל הקובע, לגבי כל שחקן שנבחר, כמה הוא צריך לשלם. כלל התשלום הוא יחיד (לפי מאירסון) והוא ערך הסף של כל משתתף.
תועלת - אם שחקן i נבחר ומשלם p_i, אז התועלת שלו היא ההפרש - $v_i - p_i$.

משפט מאירסון: מונוטוניות היא תנאי הכרחי ומספיק לגילוי-אמת. כלומר:
(א) לכל כלל-בחירה לא-מונוטוני - אין כלל-תשלום מגלה-אמת. (ב) לכל כלל-בחירה מונוטוני - קיים כלל-תשלום מגלה-אמת.
(ג) אם ישנה דרישה נוספת, לפיה מי שאינו נבחר – משלם 0, אז כלל-התשלום המגלה-אמת הוא יחיד.

VCG נגד מאירסון (שני מכרזים מגלי אמת):
היתרון של VCG - שהוא עובד גם כשהשחקנים הם רב-פרמטריים - יש להם הרבה ערכים שונים (למשל בבעיית בחירת המסעדות - לכל שחקן יש ערך לכל מסעדה). החסרון של VCG - שכלל-הבחירה היחיד שהוא יודע לעבוד איתו זה כלל "מקסום סכום הערכים".
היתרון של מאירסון - הוא עובד עם כל כלל-בחירה מונוטוני; החיסרון של מאירסון - שהוא יודע לעבוד רק עם שחקנים חד-פרמטריים - לכל שחקן יש רק מספר אחד המציין את הערך שלו ל"היבחרות". הוא לא יכול לפתור בעיות עם שחקנים רב-פרמטריים.

אלגוריתמים למקסום רווח (8):

שיקול: מקסום רווח, נרצה למצוא כלל בחירה למכרז מאירסון כך שתשלומי השחקנים יהיו מקסימלים וכך נמקסם את הרווח (מקסום רווח לא הולך יחד עם יעילות פארטו).
כשאנחנו מדברים על "מקסום רווח", הכוונה לרווח ממוצע (תוחלת הרווח). אנחנו מניחים שיש לנו מידע סטטיסטי כלשהו על התפלגות הערכים של קונים. לדוגמה, נניח שאנחנו מוכרים חפץ אחד לקונה אחד, ואנחנו יודעים לפי סקר-שוק שעשינו, שהערך שלו הוא בין 10 ל-30, עם התפלגות אחידה (סיכוי שווה לכל מספר בין 10 ל-30). עכשיו אפשר לחשב את תוחלת הרווח (ערך הסף כפול ההסתברות) של כללי-בחירה שונים:

- "בחר באופן שממקסם את סכום הערכים": ערך-הסף הוא 0, ולכן התשלום הוא 0, ולכן תוחלת הרווח היא 0.
- "בחר את הקונה אם הערך שלו לפחות 10": ערך-הסף הוא 10, הקונה תמיד נבחר (בהסתברות 1) ותמיד משלם, ולכן תוחלת הרווח היא 10.
- "בחר את הקונה אם הערך שלו לפחות 20": ערך-הסף הוא 20, הקונה נבחר בהסתברות 1/2 (כי לחצי מכל הקונים יש ערך לפחות 20), ולכן תוחלת הרווח היא 20/2 = 10.
- "בחר את הקונה אם הערך שלו לפחות 15": ערך-הסף הוא 15, הקונה נבחר בהסתברות 3/4, ולכן תוחלת הרווח היא 3/4*15 = 11.25.

חישוב הערך הוירטואלי:

1. לכל שחקן i, אנחנו מניחים שידועה לנו פונקציה F_i, המייצגת את התפלגות הערכים שלו, ומוגדרת כך: $F_i(x) = Prob[v_i < x]$
חישוב הפונקציה F_i נעשה ע"י סטטיסטיקאים וסוקרי-שוק. למשל, אם הם רואים שכל הערכים של קונים הם בין 10 ל-30 ויש בערך מספר שווה של קונים בכל אחד מהערכים האלה, אז הפונקציה תהיה: $F_i(x) = \frac{x-10}{30-10} \;\; for \;\; 30 \le x \le 10$
2. נגזור את הפונקציה הזאת ונקבל פונקציה שנקראת צפיפות הערכים (מקובל לסמן אותה ב $F_i'(x)$).
למשל, הנגזרת של הפונקציה הקודמת היא: $F_i'(x) = \frac{1}{30-10} \;\; for \;\; 30 \le x \le 10$.
3. נחשב את פונקציית הערך הוירטואלי - $r_i(x) = x - \frac{1-F_i(x)}{F_i'(x)}$.
עבור הפונקציה הקודמת נקבל: $r_i(x) = x - \frac{1-\frac{x-10}{20}}{\frac{1}{20}} = 2x - 30$.
נשים לב: הערך הוירטואלי קטן מהערך האמיתי (במקרה זה הוא מתחיל במינוס 10 ($r_i(10) = -10$) ועולה עד 30 ($r_i(30) = 30$)).

משפט: לכל כלל-בחירה, כאשר התשלומים נקבעים לפי ערכי-הסף, תוחלת הרווח שווה לתוחלת סכום הערכים הוירטואליים של הנבחרים.

- "בחר את הקונה אם הערך שלו לפחות 10": הקונה נבחר בהסתברות 1. הערך הוירטואלי שלו מתפלג אחיד בין -10 לבין 30 ולכן התוחלת שלו 10.
- "בחר את הקונה אם הערך שלו לפחות 20": הקונה נבחר בהסתברות 1/2. במקרה שהוא נבחר, הערך הוירטואלי שלו מתפלג אחיד בין 10 לבין 30. לכן התוחלת היא 1/2 מ-20 שזה 10.
- "בחר את הקונה אם הערך שלו לפחות 15": הקונה נבחר בהסתברות 3/4. במקרה זה הערך הוירטואלי שלו מתפלג אחיד בין 0 לבין 30, לכן התוחלת היא 3/4 מ-15 שזה 11.25.

למה זה עוזר לנו? כי עכשיו אנחנו יודעים איזה כלל-בחירה אנחנו רוצים: כדי למקסם רווח, כלל-הבחירה שלנו צריך להיות **בחר את הקבוצה שבה סכום הערכים הוירטואליים הוא הגדול ביותר**.

אלגוריתם למכירת חפץ אחד לקונה אחד (ממקסם רווח למוכר אך אינו מגלה אמת):
כלל-הבחירה של מאירסון במקרה זה הוא: "בחר את הקונה אם"ם הערך הוירטואלי שלו גדול מאפס"
וכלל-הבחירה הוא "בחר את הקונה אם"ם הערך שלו גדול מ-15". כלל-הבחירה מיד קובע לנו את כלל-התשלומים - הקונה יצטרך לשלם 15 (אם יזכה).

אלגוריתם למכירת חפץ אחד להרבה קונים עם אותה התפלגות (לכל הקונים יש אותה פונקציית ערך וירטואלי):
אנחנו צריכים לבחור את הקבוצה שממקסמת את סכום הערכים הוירטואליים; הקבוצה הזאת כוללת את הקונה עם הערך הגדול ביותר אם ערכו גדול מ- 15, וקבוצה ריקה אם כל הערכים קטנים מ-15. ערך-הסף של הזוכה הוא המקסימום בין 15 לבין הערך השני בגודלו. מכאן שהאלגוריתם ממקסם-הרווח במקרה זה הוא פשוט מכרז ויקרי עם מחיר-מינימום 15.
נחשב את תוחלת הרווח של מכרז זה כשיש שני קונים עם התפלגות אחידה בין 10 ל 30. למכרז יש 4 תוצאות אפשריות:

- בהסתברות 1/16, שני הערכים קטנים מ-15, והרווח הוא 0.
- בהסתברות 3/16, הערך הראשון קטן מ-15 והשני גדול מ-15 והרווח הוא 15. בהסתברות 3/16, הערך הראשון גדול מ-15 והשני קטן מ-15 והרווח הוא 15.
- בהסתברות 9/16, שני הערכים גדולים מ-15 והרווח הוא הערך הקטן יותר. ידוע שכששני ערכים מתפלגים בהתפלגות אחידה וזהה, תוחלת הרווח של הערך הקטן היא 1/3 המרחק בין הקצה הנמוך לקצה הגבוה, ושל הערך הגדל - 2/3 המרחק בין הערך הנמוך לערך הגבוה (באופן כללי, כש-n ערכים מתפלגים אחיד וזהה, תוחלת הרווח של הערך הקטן היא $\frac{n+1}{1}$ המרחק, של השני מלמטה $\frac{n+1}{2}$, וכן הלאה, של הגדול $\frac{n+1}{n}$המרחק). במקרה שלנו, כששני הערכים גדולים מ-15, שניהם מתפלגים אחיד בין 15 ל30, כך שהתוחלת של הערך הקטן מבינהם היא 20.

כשמסכמים את כל 4 המקרים מקבלים שתוחלת הרווח היא 270/16 = 16.875.

אלגוריתם למכירת חפץ אחד לשני קונים עם התפלגות שונה:
נניח שאנחנו עושים מכרז על מכונית, ויש שני קונים - אמריקאי וישראלי. לאמריקאי יש התפלגות אחידה בין 10 ל-30 ולישראלי יש התפלגות אחידה בין 20 ל-40. במקרה הזה כלל הבחירה הוא מורכב כי פונקציית הערך הוירטואלי שונה.
נסמן את הערך של האמריקאי ב-x ושל הישראלי ב-y. אז כלל-הבחירה הוא:

$$if\ 2x-30>2y-40\ and\ 2x-30>0 \rightarrow\ \ sell\ to\ American$$
$$if\ 2y-40>2x-30\ and\ 2y-40>0 \rightarrow\ \ sell\ to\ Israeli$$
$$if\ 0>2x-30\ and\ 0>2y-40\ \ \rightarrow\ \ do\ not\ sell\ at\ all$$

כלל התשלום יהיה ערך הסף (הערך שבו הזוכה עובר מהפסד לזכייה - הערך שבו הוא בדיוק ב"תיקו" עם השני, נניח $x=23, y=27$, האמריקאי נבחר וישלם 22).

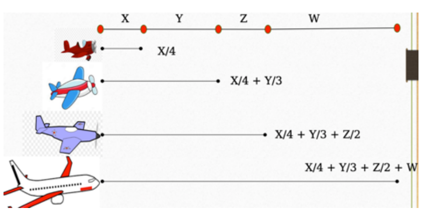
חלוקת עלויות - ערך שאפלי (9):

הבעיה: חלוקה הוגנת כשאר ההעדפות שוות (כולם רוצים כמה שיותר כסף) והזכויות שונות (נוסע אחד יורד אחרי ק״מ ונוסע שני אחרי 4 ק״מ – לא הגיוני שישלמו אותו דבר).

כיסוי מלא - כשכולם נוסעים יחד, עלות הנסיעה מכוסה במלואה ע"י הנוסעים.
הגינות (סימטריה) - תשלום שכל שחקן משלם צריך להיות תלוי רק בעלות השולית שהוא מוסיף. עקרון הסימטריה קובע, שאם לשני שחקנים, i ו-j, יש בדיוק אותה תרומה שולית לכל תת-הקבוצות שאינן מכילות אותם, אז הם צריכים לשלם את אותו תשלום בדיוק.
עקרון האפס - "שחקן אפס" הוא שחקן שאינו מוסיף שום עלות לשום תת-קבוצה שהוא מצטרף אליה. עקרון האפס קובע, ששחקן כזה לא צריך לשלם או לקבל כלום.
ליניאריות - התשלומים נמדדים באותן יחידות שבהן מודדים את הערכים. למשל, אם הערכים בשקלים אז גם התשלומים בשקלים. מכאן, אם מכפילים את כל הערכים פי 100 (או קבוע כלשהו k), ומפעילים את אותו כלל-תשלומים - מקבלים תשלומים גדולים פי 100 (או פי k).

קבוצה:	0	א	ב	ג	א,ב	א,ג	ב,ג	א,ב,ג
עלות:	0	10	15	25	20	25	30	37
סדר:	א-ב-ג	א-ג-ב	ב-א-ג	ב-ג-א	ג-א-ב	ג-ב-א	ממוצע	
א:	10	10	5	7	0	7	6.5	
ב:	10	12	15	15	12	5	11.5	
ג:	17	15	17	15	25	25	19	
סכום:	37	37	37	37	37	37	37	

אלגוריתם שאפלי (מחזיר חלוקה שהיא כיסוי מלא, סימטרית, לינארית ומקיימת את עקרון האפס):
1. עבור על כל $n!$ הסידורים האפשריים של השחקנים.
2. עבור כל סידור וכל שחקן, חשב את העלות השולית שלו בסידור זה.
3. התשלום של כל שחקן הוא הממוצע החשבוני של העלויות השוליות שלו בכל אחד מהסידורים.



אלגוריתם לבעיית שדה התעופה (n חברות-תעופה, כל אחת צריכה מסלול-המראה באורך אחר, רוצים לבנות מסלול-המראה אחד לכל החברות, והשאלה היא איך לחלק את התשלומים ביניהן?):
אפשר לחשב את ערך שאפלי בדרך הרגילה, אבל אפשר לחשב באופן יעיל יותר ע"פ העקרונות של שאפלי.
לפי עקרון הליניאריות, אפשר לפרק את הבעיה לסכום של כמה תת-בעיות:
לכל i בין 1 ל-n, נגדיר תת-בעיה שבה צריך לשלם רק על הקטע בין $i-1$ לבין i. הבעיה המקורית היא סכום של כל ה-n תת-בעיות, ולכן, אם נפתור כל בעיה בנפרד, נוכל פשוט לחבר את התוצאות ולקבל את הפתרון לבעיה המקורית.
כדי לפתור את תת-בעיה i, נשים לב שבבעיה זו, כל השחקנים הקטנים מ-i הם "שחקני אפס", ולכן לפי עקרון האפס הם לא משלמים כלום.
כל השחקנים מ-i ומעלה הם סימטריים, ולכן לפי עקרון הסימטריה כל אחד מהם משלם אותו הדבר. המסקנה היא, שבתת-בעיה i, העלות של הקטע בין $i-1$ לבין i מתחלק שווה בשווה בין השחקנים $i,\ldots,n$ כל אחד מהם משלם $\frac{1}{n-i+1}$ מעלות הקטע.

אלגוריתם לוינגר-חסון-עזריה לבעיית שיתוף נסיעות כאשר סדר הורדת הנוסעים נקבע מראש (קלט: גרף כבישים מכוון):
גם כאן, אפשר לחשב את ערך שאפלי בצורה הרגילה, אבל אפשר גם לחשב ביעילות ע"י פירוק הבעיה לסכום של תת-בעיות ($O(n^2)$ תתי בעיות):
- לכל k, נגדיר תת-בעיה שבה אנחנו משלמים רק על נסיעה ישירה מ-0 ל-k (במסלול הקצר ביותר). שימו לב שאנחנו משלמים את הנסיעה הזאת, אם ורק אם נוסע k נמצא וגם אין במונית אף נוסע שמספרו קטן מ-k. לכן:
  - כל הנוסעים שמספרם גדול מ-k הם שחקני-אפס ולא משלמים כלום.
  - ערך שאפלי של k בבעיה זו הוא המרחק הישיר בין 0 ל-k, מחולק ב-k – כי רק באחד מכל k סדרים נוסע k מופיע לפני כל הנוסעים $k-1,\ldots,1$, ורק במקרה זה העלות השולית שלו חיובית.
  - ערך שאפלי של כל נוסע $j<k$ הוא **מינוס** המרחק הישיר בין 0 ל-k, מחולק ב $k(k-1)$ – כי מתוך הסדרים שבהם k מופיע לפני כל הנוסעים $k-1,\ldots,1$, באחד מכל $k-1$ סדרים, הנוסע j הוא הראשון שמופיע מיד אחרי k, ובמקרה זה הוא גורם לכך שאנחנו לא נוסעים במסלול הישיר מ-0 ל-k, כלומר הוא מפחית את עלות המסלול.
- לכל $i<k$, נגדיר תת-בעיה שבה אנחנו משלמים רק על נסיעה ישירה מ-i ל-k. באופן דומה לחישוב למעלה:
  - כל הנוסעים שמספרם קטן מ-i או גדול מ-k הם שחקני אפס.
  - ערך שאפלי של i,k הוא המרחק הישיר בין i ל-k, מחולק ב $(k-i)(k-i+1)$.
  - ערך שאפלי של כל נוסע j שעבורו $k<j<i$ הוא **מינוס** המרחק הישיר בין i ל-k, מחולק ב $(k-i+1)(k-i)$ כפול $\frac{(k-i-1)}{2}$.

חלוקת תקציב רציפה (10):

הבעיה: נתונים n אזרחים ו-m נושאים. לכל אזרח i ונושא j, נסמן ב: $u_{i,j}$ את התועלת שאזרח i מייחס לנושא j.
לשם פשטות, אנחנו מניחים שהתועלות הן בינאריות – 0 או 1. המשמעות היא, שכל אזרח צריך רק לסמן "וי" ליד הנושאים שהוא תומך בהם.
פלט: חלוקה של התקציב בין הנושאים השונים.
אנחנו מניחים שהתועלת, שכל אזרח מפיק מווקטור d מסויים, שווה לסכום התקציב המועבר לנושאים שהוא תומך בהם:$u_{i(d)} = sum[j]\ u_{i,j} *\ d_j$

דרישות:

- תקציב הוגן (חלק הוגן לקבוצות) - לכל קבוצת-אזרחים K בגודל k, יש לתת לפחות k/n מהתקציב לנושאים, שלפחות אחד מחברי-הקבוצה K תומך בהם. ניתן להוכיח.
  מתכונה זו נובעות התכונות הבאות:
  - *חלק הוגן ליחידים/פרופורציונאליות:* התועלת של כל אזרח היא לפחות$\frac{1}{n}$ מהתועלת הכללית. כלומר: לכל אזרח i, התקציב הכולל המועבר לנושאים שהוא תומך בהם צריך להיות לפחות C/n.
  - *חלק הוגן לקבוצות אחידות:* כל קבוצת-אזרחים K בגודל k, שהדעות שלהם זהות (= הם תומכים באותם נושאים), יש לתת לפחות k/n מהתקציב לנושאים הנתמכים על-ידי חברי הקבוצה K.
- יעילות - תקציב d נקרא יעיל פארטו אם לא קיים תקציב אחר, הנותן לכל האזרחים תועלת גדולה באותה מידה, ונותן לאזרח אחד לפחות תועלת גדולה יותר.

האלגוריתם האוטיליטרי לחלוקת תקציב (יעיל פארטו ומגלה אמת אך לא הוגן):
$\max[d] \;\; sum[i] \;\; u_i(d)$ (פשוט מעבירים את כל התקציב C רק לנושא/ים, שמספר התומכים בהם הוא הגדול ביותר)

אלגוריתם אנארכי לחלוקת תקציב (פריק ולכן הוגן לקבוצות, מגלה אמת אך אינו יעיל פארטו):
כל אזרח מחלק את החלק שלו שווה בשווה בין הנושאים שהוא תומך בהם.

וקטור-תקציב d נקרא **פריק**, אם ניתן לבצע אותו באופן הבא:
נותנים C/n לכל אזרח, ואומרים לו כמה בדיוק להעביר לכל אחד מהנושאים שהוא תומך בהם, כך ששסכום ההעברות שווה בדיוק לווקטור d. הגדרה שקולה: תקציב פריק הוא תקציב שניתן להציג ע"י מספרים $d_{i,j}$ כך ש:

$$For\ every\ j\ in\ 1,\dots m: \quad sum[i=1,\dots,n]\ d_{i,j} = d_j$$
$$For\ every\ i\ in\ 1,\dots n: \quad sum[j=1,\dots,m]\ d_{i,j} = C/n$$
$$For\ every\ i,j: \quad d_{i,j} > 0\ only\ if\ u_{i,j} > 0$$

הערה: יתרון גדול של תקציב פריק הוא **שקיפות** – כל אזרח יכול לוודא, שהחלק שלו בתקציב אכן מועבר לנושאים שהוא תומך בהם.
משפט: תקציב d נותן חלק הוגן לקבוצות אם ורק אם הוא פריק.
משפט: כל תקציב הממקסם את סכום הלוגריתמים הוא פריק (ולכן גם נותן חלק-הוגן לקבוצות).

אלגוריתם Nash לחלוקת תקציב (הוגן לקבוצות + יעיל פארטו, אינו מגלה אמת):
אלגוריתם נאש מוצא וקטור-תקציב הממקסם את מכפלת התועלות של כל האזרחים. באותיות: $\max[d] \;\; product[i]\ u_i(d)$
כידוע, סכום של לוגריתמים שווה לוגריתם של המכפלה.
לכן, דרך שקולה לתאר את אלגוריתם נאש היא מקסום סכום הלוגריתמים של התועלות: $\max[d] \;\; sum[i] \;\; \log{(u_i(d))}$
מקסום סכום של לוגריתמים זו בעיית אופטימיזציה קמורה, ולכן ניתן לפתור אותה ביעילות, למשל בעזרת הספרייה cvxpy.

משפט: לא קיים אלגוריתם המקיים בו-זמנית את התכונות הבאות: (1) יעיל-פארטו. (2) מגלה-אמת. (3) הוגן – אפילו ליחידים.
(4) אנונימי - לא מושפע משינוי הסדר בין האזרחים (כל האלגוריתמים שראינו למעלה הם אנונימיים, אלגוריתם לא אנונימי – אבן פז).
(5) ניטרלי – לא מושפע משינוי הסדר בין הנושאים (כל האלגוריתמים שראינו למעלה הם ניטרליים).

האלגוריתם האוטיליטרי-על-תנאי (הוגן-לקבוצות, מגלה-אמת, ומחזיר תקציב יעיל-פארטו **בקבוצת התקציבים הפריקים** – אין שיפור פארטו פריק):
מחשב וקטור, שבו סכום התועלות הוא הגדול ביותר תחת האילוץ שכל אזרח תורם רק לנושאים שהוא תומך בהם.
איך מוצאים אותו? - אומרים לכל אזרח i להשקיע את כל החלק שלו בתקציב בנושאים, שמספר התומכים בהם הוא הגדול ביותר, מבין הנושאים שהוא תומך בהם. (אם תמי תומכת בנושאים ׳א׳ ו ׳ב׳, אבל יש עוד 10 אזרחים התומכים ב ׳א׳ ועוד 20 אנשים התומכים ב ׳ב׳, אז האלגוריתם יגיד לתמי להשקיע את כל התקציב שלה ב ׳ב׳.)

תקצוב השתתפותי:
פרופורציונליות חזקה - נסמן את הסכום הקצוב באות L, ואת מספר האזרחים המצביעים באות n.
תקציב נקרא פרופורציונלי-חזק אם לכל קבוצת מצביעים בגודל k (מתוך n): אם כל חברי הקבוצה מסכימים על פריטים שהעלות הכוללת שלהם לפחות $\frac{kL}{n}$ - אז הסכום המוקצב לפריטים, שלפחות אחד מחברי-הקבוצה רוצה, הוא לפחות $\frac{kL}{n}$.
פרופורציונליות רגילה - תקציב נקרא פרופורציונלי אם לכל קבוצת מצביעים בגודל k (מתוך n): הסכום המוקצב לפריטים, שלפחות אחד מחברי-הקבוצה רוצה, הוא לפחות העלות הגדולה ביותר של קבוצת-פריטים שכל חברי הקבוצה מסכימים עליהם, ועולה לכל היותר kL/n

אלגוריתם עזיז-לי-טלמון למציאת תקציב (מחזיר תקציב פרופורציונלי רגיל ולא חורג מגבולות התקציב. זמן ריצה $(O(n*2^m)$:
האלגוריתם מחזיק שני משתנים: משתנה אחד נקרא "תקציב" ומכיל את קבוצת הפריטים שהחלטנו לממן (מאותחל לקבוצה ריקה); משתנה שני נקרא "מקופחים" ומכיל את קבוצת האזרחים, שהפריטים שהם רוצים עדיין לא נכנסו לתקציב (מאותחל לקבוצת כל האזרחים).
אתחול: תקציב := קבוצה ריקה. מקופחים := כל האזרחים.
נסמן באות m את מספר-הפריטים הכולל.
סדר את 2^m קבוצות הפריטים בסדר יורד של עלות. לכל קבוצת-פריטים Y, מהיקרה לזולה:

- חשב את קבוצת-האזרחים K שרוצים את כל הפריטים ב-Y. נניח שבקבוצה יש k אזרחים.
- אם עלות הפריטים ב-Y היא לכל היותר kL/n - הוסף את הפריטים ב-Y לתקציב. והורד את האזרחים ב-K מקבוצת המקופחים.

תקצוב השתתפותי – מיזוג הצעות תקציב (11)

הבעיה: נניח, שהאזרחים לא מסתפקים בסימון "וי" על נושאים שהם תומכים בהם: הם רוצים לקבוע את כל התקציב, ולהחליט כמה כסף בדיוק ילך לכל נושא. השאלה היא, איך אפשר למזג את כל התקציבים האלו, ולהגיע לתקציב המייצג כולם באופן הוגן?

הקלט: הכסף בקופה – C , נושאים (סעיפי התקציב) - $1,\dots,m$, האזרחים - $1,\dots,n$ ווקטור התקציב האידאלי לכל אזרח i - $(p_{i,1},\dots,p_{i,m})$ כאשר $p_{i,1}+\dots+p_{i,m}=C$.

הפלט: וקטור d המייצג תקציב - $(d_1,\dots,d_m)$ כך ש$d_1+\dots+d_m=C$.

התועלת של אזרח i מהתקציב d היא: $u_i(d)=-sum[j=1,\dots,m]\ |d_j-p_{i,j}|$

אלגוריתם לחלוקת סעיף אחד (קלט: כמה כסף כל אזרח רוצה שילך לתקציב, לא מגלה אמת): לקחת את הממוצע של כל ההצבעות.

אלגוריתם לחלוקת סעיף אחד (קלט: כמה כסף כל אזרח רוצה שילך לתקציב, לא יעיל פארטו):
להתעלם מההצבעות של האזרחים ולבחור מספר קבוע באופן שרירותי.

אלגוריתם לחלוקת סעיף אחד (קלט: כמה כסף כל אזרח רוצה שילך לתקציב, לא הוגן כיוון שלא מקיים אנונימיות):
להפעיל את "אלגוריתם הדיקטטור" ולתת לאזרח אחד לבחור את התקציב שהוא רוצה.

אלגוריתם החציון (אנונימי, יעיל פארטו, מגלה אמת והוגן כשיש סעיף אחד. שימושי לא רק להחלטה על גודל התקציב, אלא לכל נושא חד-ממדי):
1. סמן את הצבעותיהם של האזרחים ב: $p_1,\dots,p_n$.
2. סדר את כל הצבעות האזרחים בסדר עולה, כך ש: $p1\ \le\dots\le\ pn$.
3. בחר את הצבעה מספר $n/2$ (עגל כלפי מעלה).

הוגן לקבוצות: אלגוריתם לקביעת תקציב נקרא הוגן לקבוצות אם, כאשר כל אזרח רוצה לתת 100% מהתקציב לנושא אחד מסויים, חלוקת התקציב בין הנושאים נקבעת באופן יחסי למספר האזרחים התומכים בנושא זה.

אלגוריתם החציון המוכלל ל2 סעיפים (אנונימי, מגלה אמת והוגן לקבוצות אם הפונקציות ליניאריות, יעיל פארטו רק אם יש לכל היותר $n-1$ הצבעות קבועות):
1. נבחר מראש קבוצה כלשהי של מספרים $f_1,\dots,f_k$, שייקראו ההצבעות הקבועות.
2. נוסיף את קבוצת ההצבעות הקבועות לקבוצת ההצבעות של האזרחים, כך שיהיו לנו בסך-הכל $n+k$ הצבעות.
3. נפעיל את אלגוריתם החציון המקורי על $n+k$ ההצבעות האלו.

אלגוריתם החציון המוכלל לm סעיפים (אנונימי, מגלה אמת, הוגן לקבוצות (רק במקרה קיצוני שבו האזרחים מחולקים לקבוצות הנותנות 100% מהתקציב לנושא אחד, אם זה לא המצב אפשר לקחת את ממוצע התקציבים אבל אז הוא לא מגלה אמת) , לא יעיל פארטו):

נגדיר $n-1$ פונקציות: $f_1(t),\dots,f_{n-1}(t)$, כך ש: $f_i(t)=C*\min(1,i*t)$

כל פונקציה i היא רציפה, עולה, ומקיימת: $f_i(0)=0,f_i(1)=C$. לכל t בין 0 ל-1, ניתן להריץ את אלגוריתם החציון המוכלל עם הצבעות קבועות $f_1(t),\dots,f_{n-1}(t)$ כיוון שהפונקציות עולות, גם החציון יהיה פונקציה עולה של t:

- עבור $t=0$, יהיו לנו $n-1$ הצבעות קבועות ב-0, ולכן החציון בכל נושא j יהיה הערך הנמוך ביותר שאזרח כלשהו ציין עבור נושא j. לכן סכום החציונים יהיה לכל היותר C.
- עבור $t=1$, יהיו לנו $n-1$ הצבעות קבועות ב-C, ולכן החציון בכל נושא j יהיה הערך הגבוה ביותר שאזרח כלשהו ציין עבור נושא j. לכן סכום החציונים יהיה לפחות C.
- הפונקציות רציפות, ולכן לפי משפט ערך הביניים יהיה t^* כלשהו בין 0 ל-1, שעבורו סכום החציונים יהיה בדיוק C.
אנחנו נבחר t^* כזה ונקבע את התקציב בעזרת אלגוריתם החציון המוכלל עם הצבעות קבועות $f_1(t),\dots,f_{n-1}(t)$.
ניתן למצוא את t^* בעזרת חיפוש בינארי.

האם קיים אלגוריתם המקיים את שלושת התכונות: גילוי-אמת, הגינות לקבוצות, ויעילות-פארטו? התשובה היא לא.
האם קיים אלגוריתם מגלה-אמת כלשהו, שגם מבטיח הגינות חזקה לקבוצות? נכון לרגע זה, אנחנו לא יודעים.

אלגוריתמי החלפות ומעגלים (12):

הבעיה: נניח שכמה עובדים משובצים לתורניות, וחלק מהם היו מעדיפים להתחלף עם אחרים. או, כמה סטודנטים משובצים לחדרים במעונות והיו מעוניינים להתחלף. אנחנו רוצים ליצור שיבוץ חדש שיהיה יעיל פארטו.

מעודד השתתפות - אלגוריתם נקרא מעודד השתתפות, אם כל משתתף מעדיף את תוצאת האלגוריתם על-פני המצב לפני האלגוריתם (או לפחות אדיש בין התוצאות).

קבוצה מערערת - קבוצה היכולה לפרוש מהמערכת ולבצע מסחר פנימי בתוך הקבוצה, והתוצאה שתתקבל תהיה טובה יותר לפחות לאחד מחברי הקבוצה, ולא פחות טובה לכל שאר חברי הקבוצה.

שיבוץ יציב - בהינתן שיבוץ מסויים של אנשים לבתים, אם אין קבוצה מערערת, השיבוץ נקרא יציב. שיבוץ יציב הוא יעיל פארטו ומעודד השתתפות.

אלגוריתם מעגלי המסחר (מעודד השתתפות, מגלה אמת, מוצא שיבוץ יעיל פארטו יציב ומסתיים תמיד. זמן ריצה-$O(n^2)$):

קלט: שיבוץ של אנשים לבתים - לכל איש יש בית אחד.

האלגוריתם מחזיק גרף מכוון: הצמתים הם האנשים והבתים. יש קשת מכל אדם אל הבית שהוא הכי רוצה, ומכל בית אל האדם שהוא שייך אליו.

מעדכנים את הגרף באופן הבא: כל עוד הגרף לא ריק:
1. מוצאים מעגל מכוון בגרף (למשל ע"י אלגוריתם DFS).
2. מבצעים את ההחלפה במעגל: כל אדם מקבל את הבית שהוא מצביע עליו.
3. מוחקים מהגרף את הצמתים של האנשים והבתים שהשתתפו בהחלפה.
4. לכל איש שנשאר בגרף, מעדכנים את הקשת שלו כך שתצביע לבית שהוא הכי רוצה מאלה שנשארו וחוזרים לשלב 1.

<u>הערה</u>: אם כל יחסי ההעדפה חזקים (אין אדישות), אז אלגוריתם מעגלי המסחר מוצא שיבוץ יציב. ויש רק שיבוץ יציב אחד - והוא זה שמחזיר אלגוריתם מעגלי המסחר. אם יש אדישות האלגוריתם יכול לבחור שרירותית איזה בית לתת אבל אז התוצאה לא תהיה יעילה פארטו.

החלפות כשיש אדישות:

<u>רכיב קשיר חזק</u>: בגרף מכוון, רכיב קשיר חזק הוא אוסף של צמתים, שאפשר להגיע מכל אחד מהם לכל אחד מהם במסלול מכוון.
<u>רכיב קשיר חזק סופי</u>: רכיב-קשיר-חזק C נקרא סופי אם: (1) אין קשתות יוצאות משחקנים ב-C לרכיבים אחרים.
(2) יש קשת מכל שחקן ב-C לעצמו. במילים אחרות: כל שחקן ב-C מחזיק באחד הבתים שהוא הכי רוצה, מבין הבתים שיש עכשיו בגרף.

<u>אלגוריתם מעגלי המסחר עם אדישות (מחזיר שיבוץ חלש, מגלה אמת ויעיל פארטו):</u>
<u>קלט</u>: שיבוץ של אנשים לבתים - לכל איש יש בית אחד.
האלגוריתם מחזיק גרף מכוון: הצמתים הם האנשים והבתים. יש קשת מכל אדם אל הבית שהוא הכי רוצה, ומכל בית אל האדם שהוא שייך אליו.

1. כל עוד יש בגרף רכיב-קשיר-חזק סופי C (DFS יודע למצוא רכיב קשירות חזק בגרף):
   - נותנים לכל השחקנים ב-C את הבית שהם מחזיקים בו עכשיו.
   - מוציאים מהשוק את השחקנים ב-C ואת הבתים שלהם.
   - מעדכנים את ההעדפות של השחקנים האחרים, ובונים מחדש את הגרף.
2. אם אין בגרף רכיב-קשיר-חזק סופי:
   - משתמשים בכלל-בחירה מסויים כדי לבחור, לכל שחקן, בית אחד מבין הבתים שהוא הכי רוצה.
   - מוצאים מעגל מכוון בגרף שנוצר.
   - מבצעים את ההחלפה במעגל (אבל משאירים בשוק את השחקנים והבתים).
3. חוזרים על סעיפים 1,2 עד שהגרף ריק.

ישנם כמה כללי-בחירה שאפשר להשתמש בהם בסעיף 2. נציג כלל-בחירה אחד – **כלל סבן-סתורמן**.
כדי להגדיר את הכלל, נשתמש במושגים הבאים:
<u>שחקן מקנא</u> – שחקן, שהבית שהוא מחזיק בו כעת, אינו אחד מהבתים שהוא הכי רוצה (כלומר: אין קשת ממנו לביתו).
<u>שחקן מסודר</u> – שחקן שכבר בחר, מבין הבתים שהוא הכי רוצה, בית אחד שעליו הוא יצביע.
<u>בית עדיף</u> – אנחנו מגדירים מראש סדר-עדיפות קבוע על הבתים; בית "עדיף" הוא בית שהעדיפות שלו גבוהה ביותר.
בכל שלב, הכלל בוחר בתים לשחקנים באופן הבא:

- כל שחקן מקנא בוחר, מבין הבתים שהוא הכי רוצה, את הבית הכי עדיף.
  עכשיו, כל השחקנים המקנאים הם מסודרים.
- מבין השחקנים הלא-מסודרים, מסתכלים על השחקנים, שאחד הבתים שהם הכי רוצים שייך לשחקן מסודר.
  מבין השחקנים האלה, בוחרים את זה שמחזיק בבית הכי עדיף.
  מבין הבתים שהוא הכי רוצה, השייכים לשחקן מסודר, בוחרים עבורו את הבית הכי עדיף.
  ממשיכים באופן זה עד שכל השחקנים מסודרים.

כאשר כל השחקנים מסודרים, לכל שחקן יש קשת יוצאת אחת בדיוק, ולכן קיים מעגל מכוון, ואפשר להמשיך בשלב ב באלגוריתם.

<u>משפט</u>: אלגוריתם מעגלי המסחר עם אדישות עם כלל-הבחירה של סבן-סתורמן מסתיים תוך $2n$ צעדים לכל היותר.
<u>משפט</u>: אלגוריתם מעגלי המסחר עם אדישות מחזיר תמיד שיבוץ יעיל-פארטו.
<u>הערה</u>: אלגוריתם מעגלי המסחר עם אדישות מחזיר שיבוץ יעיל-פארטו לכל כלל-בחירה (בתנאי שהאלגוריתם מסתיים).

<u>קבוצה מערערת-חזק</u> - בהינתן שיבוץ מסויים של אנשים לבתים, נגדיר קבוצה מערערת-חזק כקבוצה היכולה לפרוש מהמערכת ולבצע מסחר פנימי בתוך הקבוצה, והתוצאה שתתקבל תהיה טובה יותר לכל חברי הקבוצה. אם אין קבוצה מערערת-חזק, השיבוץ נקרא **יציב חלש**.
קל לראות, שכל שיבוץ יציב-חלש הוא מעודד-השתתפות.

<u>משפט</u>: אלגוריתם מעגלי המסחר עם אדישות מחזיר תמיד שיבוץ יציב-חלש.
<u>משפט</u>: אלגוריתם מעגלי המסחר עם אדישות הוא מגלה אמת (ההוכחה היא מעבר להיקף הקורס).