פינת החודש: חידות ובעיות – ללמוד וללמד קצת אחרת
כתבה וערכה: שרה פולק

אחת המטרות המרכזיות בהוראת מדעי המחשב היא הקניית היכולת לפתרון בעיות מעשיות באמצעות המחשב. אולם, במקרים רבים תלמידים "רצים לכתוב תוכנית במחשב" ומדלגים על השלבים הראשונים והחשובים בפיתוח בעיה, כמו ניתוח ותכנון. באתר זה נציע פעילויות למידה הכוללות שימוש בחידות ובעיות מאתגרות, שבפתרונן נדרש ניתוח וחקר של מצבים ותופעות, יכולת קישור בין עצמים, חשיבה מקורית ובקרה עצמית. ההתמודדות עם "פיצוח" החידות, יכול לגרות את הדמיון והשכל, ולספק לתלמיד את ההזדמנות לעסוק בניתוח בעיה, תכנון פתרון ולסיום לפתח אלגוריתם מתאים. בנוסף, שילוב בעיות מסוג זה בהוראת המקצוע, מוסיף עניין וגיוון לתהליך הלמידה.

באתר יוצגו (אחת לחודש) חידה או בעיה (לעיתים זו סדרה מתפתחת של בעיות), שבעזרתן ננסה לעודד מיומנויות ניתוח, תכנון ובקרה עצמית. כל חידה ובעיה שתוצג באתר זה תלווה בפעילויות הבאות:

1. ניתוח וחיפוש פתרונות לחידה באמצעות פורום של מורים
2. תכנון ופיתוח אלגוריתם (או אלגוריתמים) מתאים
3. כלים דידקטיים והמלצות לשילוב סוג בעיות זה (ורעיונות נוספים בנושא) בהוראת מדעי המחשב.

האתר פתוח לכל המורים במדעי המחשב והוא ישמש כבמה לדיון פתוח בפתרון החידות והבעיות, להחלפת מידע, רשמים וחוויות בדרך לפתרון הבעיה וביישומם בכיתה. כמו כן זהו המקום לחידות ובעיות שיוצגו על ידי המורים ויתווספו למאגר הפעילויות.

תוכן:

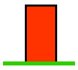
פינת דצמבר 2003 2
פינת ינואר 2004 10
פינת פברואר 2004 14
פינת מרץ 2004 19
פינת אפריל 2004 23
פינת מאי 2004 30

פינת דצמבר 2003

בעיה א:

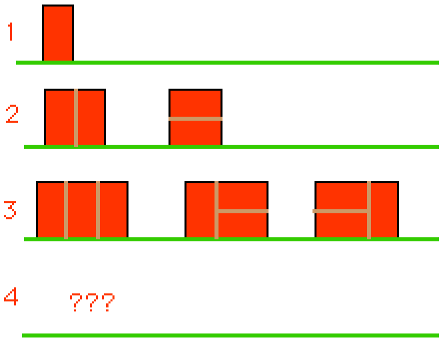
יש לבנות קיר של לבנים מלבנים סטנדרטיות בהן האורך הוא כפול מהרוחב (האורך הוא 2 יחידות והרוחב היא יחידה אחת).

גובה הקיר הדרוש הוא שתי יחידות.



בהינתן מספר לא מוגבל של לבנים מסוג זה, ברצוננו לחשב את מספר האפשרויות לבניית קיר באורך n. לדוגמה:

- לבניית קיר באורך 1 יחידה יש רק אפשרות אחת לבניית הקיר בה לבנה אחת.
- לבניית קיר באורך 2 יחידות, דרושות שתי לבנים ויש שתי אפשרויות לבניית קיר.
- לבניית קיר באורך 3 יחידות דרושות שלוש אבנים ויש שלוש אפשרויות לבניית הקיר.

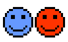
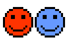
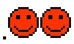
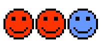
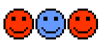
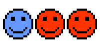
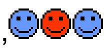
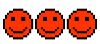


- בכמה צורות אפשר לבנות קיר באורך 4 יחידות?
- בכמה צורות אפשר לבנות קיר באורך 5 חידות?
- בכמה צורות אפשר לבנות קיר באורך n יחידות?
- מה מספר הלבנים הדרוש לבניית קיר באורך עד 10 יחידות?

בעייה ב:

קבוצת תלמידים המכילה בנים  ובנות  משתתפים באירוע בבית ספר. המארגנים רוצים להושיב חלק המשתתפים על שורה של כסאות כך ששני בנים לא ישבו בסמוך זה לזה. השאלה היא בכמה אפשרויות ניתן לסדר ב- n כסאות קבוצה של בנים ובנות בשורה כך ששני בנים לא ישבו בסמוך זה לזה.

לדוגמה:

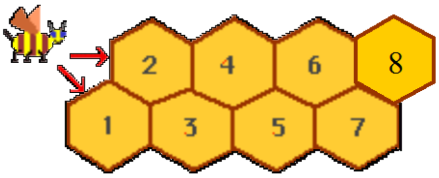
- בכסא אחד ניתן להושיב בן  או בת  כלומר 2 אפשרויות
- בשני כסאות יש 3 אפשרויות: בן ובת  או בת ובן  או שתי בנות .
- בשלוש כסאות יש 5 אפשרויות: , , , , 

- מהן האפשרויות לסידור ב- 4 כסאות קבוצת בנים ובנות כך ששני בנים לא ישבו זה ליד זה ?

- מהן האפשרויות לסידור ב- 5 כסאות קבוצת בנים ובנות כך ששני בנים לא ישבו זה ליד זה ?
- מהן האפשרויות לסידור ב- n כסאות קבוצת בנים ובנות כך **ששני** בנים לא ישבו זה ליד זה ?
- מהן האפשרויות לסידור ב- 1, 2, 3, 4 כסאות קבוצת בנים ובנות כך **ששלושה** בנים לא ישבו זה ליד זה ?

בעייה ג:

בכמה דרכים יכולה דבורה לעוף לתא מסוים אם היא מתחילה משמאל לימין (החל מתאים 1 או 2).
הדבורה מתקדמת מתא לתא סמוך כך שתמיד היא עפה מתא לתא שמספרו גדול ממנו (לדוגמה דבורה יכול לעוף מתא 2 לתא 3 אך לא להיפך).



לדוגמה:

- לדבורה יש רק דרך אחת להגיע לתא מס. 1.
- לדבורה יש שתי דרכים להגיע לתא מס. 2: דרך תאים 12 או דרך תא מס. 2
- לדבורה יש שלוש דרכים להגיע לתא מס. 3: 13 או 23 או 123

* בכמה דרכים יכולה הדבורה לעוף כדי להגיע לתא מס. 4?

* בכמה דרכים יכולה הדבורה לעוף כדי להגיע לתא מס. 8?

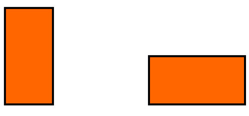
* בכמה דרכים יכולה הדבורה לעוף כדי להגיע לתא מס. 7 או מס. 8?

* בכמה דרכים יכולה הדבורה לעוף כדי להגיע לתא מס. n?

פתרון בעיה א: מה מספר האפשרויות לבניית קיר בגובה 2 ?

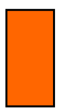
נגדיר f(n) שפירושו מספר האפשרויות לבניית קיר בגובה 2 ובאורך n עם לבנים שאורכם 2 יחידות ורוחבם 1 יחידה.

כמובן שניתן לשנות את אופן הנחתם:

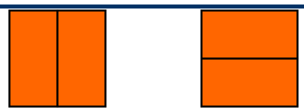


נאמת את שני המקרים הפשוטים ביותר על ידי כך שניצור את כל האפשרויות הקיימות.

* עבור n=1 קל לראות כי $f(1) = 1$ משום שכדי לקבל קיר בגובה 2 יש רק אפשרות אחת לשים את הלבנה:



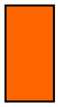
* עבור n=2 קל לראות כי $f(2) = 2$ משום שכדי לקבל קיר בגובה 2 יש רק שתי אפשרויות:



כעת נחשב את המקרה כללי $f(n)$ כאשר $n > 2$:

יש שתי אפשרויות להתחיל את בניית הקיר

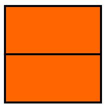
* ניתן להתחיל את בניית הקיר ולשים לבנה עומדת:



וכדי להשלים את הקיר לאורך n יש צורך לבנות קיר באורך $n-1$ כדי להשלים את הקיר לאורך n.

כלומר, יש למצוא כמה אפשרויות יש לבניית קיר באורך $n-1$? יש $f(n-1)$ אפשרויות (זו ההגדרה של $f(n)$).

* ניתן להתחיל את בניית הקיר ולשים שתי לבנים שוכבות:



וכדי להשלים את הקיר לאורך n יש צורך לבנות קיר באורך $n-2$ כדי להשלים את הקיר לאורך n.

כלומר, יש למצוא כמה אפשרויות יש לבניית קיר באורך $n-2$?

לפי ההגדרה של הפונקציה, יש $f(n-2)$ אפשרויות.

מאחר ואלו שתי האפשרויות היחידות להתחיל לבנות קיר בגובה 2,

נוכל לומר כי כדי לבנות קיר באורך $n > 2$, יש $f(n) = f(n-1) + f(n-2)$ אפשרויות, כאשר $f(1)=1$ ו- $f(2)=2$.

וזו הנוסחה של סדרת פיבונצי.

נשאר לנו לענות על השאלה: מה מספר הלבנים הדרוש לבניית קיר באורך עד 10 יחידות?

התשובה היא:

$$f(1)+f(2)+f(3)+f(4)+\ldots+f(10)$$

עם קצת משחק מתמטי נקבל כי:

$$f(2) = f(3) - f(1)$$
$$f(3) = f(4) - f(2)$$
$$f(4) = f(5) - f(3)$$
$$f(5) = f(6) - f(4)$$
$$f(6) = f(7) - f(5)$$
$$f(7) = f(8) - f(6)$$
$$f(8) = f(9) - f(4)$$
$$f(9) = f(10) - f(8)$$

$$f(10) = f(11) - f(9)$$

$$f(1)+f(2)+\ldots+f(10) = f(11) + f(10) - f(2) - f(1)$$

אבל $f(1)= 0$ ו- $f(12)=f(11)+f(10)$ ולכן:

$$f(1)+f(2)+\ldots+f(10)= f(12) - f(2)$$

זה נכון באופן כללי:

$$\sum_{1}^{n} f(n) = f(n+2) - f(2)$$

פתרון בעייה ב - סעיף 1

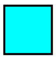
מה מספר האפשרויות להושיב n בנים ובנות כך ששני בנים לא ישבו זה ליד זה?

באופן דומה לפתרון החידה הקודמת,

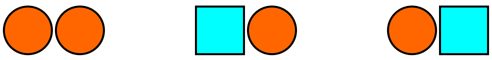
נגדיר $f(n)$ שפירושו מספר האפשרויות להושיב בנים ובנות על n כסאות כך ששני בנים לא ישבו זה ליד זה.

את המקרים הפשוטים נוכיח על ידי בניית כל האפשרויות:

* **עבור כיסא אחד**, יש $f(1)=1$ אפשרויות:

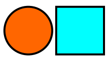
 בת  או בן

* **עבור שני כיסאות**, יש $f(2)=3$ אפשרויות:



נחשב את **המקרה הכללי** – $f(n)$ עבור $n>2$ כיסאות:

* אם בחרנו להושיב על הכיסא הראשון בן, נוכל להושיב לידו רק בת:



וכך סידרנו את היושבים ב-2 הכיסאות הראשונים.

אם כך, נשאר לנו לסדר את שאר n-2 הכיסאות, עבורם יש $f(n-2)$ אפשרויות.

* אם בחרנו להושיב על הכסא הראשון בת, נוכל לידה להושיב או בן או בת.

כלומר נשאר לנו לסדר עוד n-1 כיסאות, עבורם יש $f(n-1)$ אפשרויות.

מכאן מספר האפשרויות להושיב על n כיסאות בנים ובנות כך ששני בנים לא ישבו זה ליד זה הוא: $f(n) = f(n-1)+f(n+2)$, כאשר $f(1)=2$ ו- $f(2)=3$.

שוב סדרת פיבונצ'י, הפעם הערכים התחיליים הם בעצם האיבר השני והשלישי בסדרת פיבונצ'י.

פתרון בעייה ב - סעיף 2

מה מספר האפשרויות להושיב n בנים ובנות כך ששלושה בנים לא ישבו זה ליד זה?

גרסה אחרת של הבעיה, שהפתרון לה אינה סדרת פיבונצ'י אך ניתן לפתרון באותה דרך היא השאלה:

מה מספר האפשרויות להושיב בנים ובנות על n כסאות כך <u>שלושה</u> בנים לא ישבו זה ליד זה?

כלומר, נגדיר f(n) כמספר האפשרויות להשיב בנים ובנות על n כיסאות, כך ששלושה בנים לא ישבו זה ליד זה.

נחשב את המקרים הפשוטים עבור n=1,2,3 :

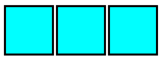
א. כמו בבעיה הקודמת, עבור **n=1** יש f(1)=2 אפשרויות.

ב. עבור **n=2** יש f(2)=4 אפשרויות, כי לשני בנים מותר לשבת זה ליד זה.

כלומר כל האפשרויות הבאות חוקיות:

ג. עבור **n=3** יש f(3)=7 אפשרויות,

כי יש 8 אפשרויות לסדר 3 בנים ובנות ורק האפשרות הבאה אינה חוקית:



עבור כל n>3 כסאות, נפעיל את שיקולים דומים לשיקולים שהפעלנו בסעיף הראשון:

* אם הושבנו בת על הכיסא הראשון, נשאר לסדר עוד n-1 כיסאות, עבורם יש f(n-1) אפשרויות.
* אם הושבנו בן על הכיסא הראשון, על הכיסא השני ניתן להושיב בן או בת:
  * אם הושבנו על הכיסא השני בת, נותרו עוד n-2 כיסאות לסידור עבורם יש f(n-2) אפשרויות.
  * אם הושבנו על הכיסא השני בן, על הכיסא השלישי נוכל להושיב רק בת, ולכן נשארו עוד n-3 כיסאות לסידור עבורם יש f(n-3) אפשרויות.

כלומר, מספר האפשרויות להושיב על n כיסאות בנים ובנות כך ששלושה בנים לא ישבו זה ליד זה הוא: f(n)= f(n-1) + f(n-2) + f(n-3)

פתרון בעייה ג: בכמה דרכים דבורה יכולה להגיע מתחילת הכוורת עד לתא n ?

קל לראות כי פתרון לשאלה זו דומה לשתי השאלות הקודמות וגם כאן התשובה היא סדרת פיבונצי.

נגדיר f(n) כמספר הדרכים שדבורה יכולה להגיע מתחילת הכוורת (תאים 1 או 2) לתא n.

קל לראות כי עבור המקרים הפשוטים :

להגיע לתא מספר 1 יש $f(1) = 1$ דרכים ולתא מספר 2 יש $f(2) = 2$ דרכים.

לכל תא בכוורת יש שני תאים סמוכים. כלומר, כדי להגיע לתא $n > 2$:
ניתן להגיע אליו ישירות מהתא הסמוך לו כלומר תא שמספרו $n-1$ עבורו יש $f(n-1)$ דרכים.
או שניתן להגיע אליו ישירות מהתא $n-2$ (שגם הוא סמוך לו), ועבורו יש $f(n-2)$ דרכים.
סך הכל לתא שמספרו $n>2$, יש $f(n) = f(n-1) + f(n-2)$ דרכים, שוב סדרת פיבונצי.

שאלה נוספת המוצגת בבעיה זו היא: בכמה דרכים יכולה הדבורה להגיע לתא מספר 7 או מספר 8?
נדגיש כי הכוונה היא להגיע לתא מספר 7 ולעצור <u>או</u> להגיע לתא מספר 8,
כלומר, לא ניתן במקרה זה לעבור מתא מספר 7 לתא מספר 8.
פתרון:
לתא מספר 7 ניתן להגיע ב- $f(7)$ דרכים.
במקרה זה לתא מספר 8 ניתן להגיע מתא מספר 6 ולכן יש $f(6)$.
סך הכל מספר הדרכים להגיע לתא מספר 7 <u>או</u> לתא מספר 8 הוא: $f(7) + f(6) = f(8)$.

פעילויות בכיתה עם חידת דצמבר 2003

רמת קושי

* השאלה השלישית (הכוורת והדבורה) היא הקלה ביותר לפתרון ומתאימה לכל הרמות.
* השאלה השניה היא הקשה ביותר לפתרון וניתן לתת אותה כבעיה שלישית. את סעיף השני בשאלה השניה ניתן להציע כאתגר לתלמידים ברמה גבוהה.

הפעלה בכיתה:

א. פעילות בנושא לולאה או רקורסיה:

* ניתן להציע לתלמידים את כל הבעיות (בבת אחת או אחת אחרי השניה) או רק חלק מהן. לאחר שהתלמידים מתמודדים עם הפתרון, ניתן לנהל דיון בכיתה על הפתרונות המוצעים, על האסטרטגיות שהם הפעילו כדי לפתור אותן. רצוי להקפיד על מתן הסברים מנומקים (או הוכחות) ולא רק דוגמאות מספריות.
* לאחר הפתרון, קישור התלמידים מתבקשים לפתח וליישם תוכנית המאפשרת לחשב את מספר האפשרויות עבור n.

ב. פעילות בנושא הפשטה:

מציאת המשותף לכל השאלות דומה לניסוח שאלה כללית (או שאלה שקולה) ומאפשר להדגים את עקרון ההפשטה במדעי המחשב. למעשה הפתרון של כל הבעיות, למרות שהן נראות שונות זו מזו, הוא סדרת פיבונצי. המטרה בכל שאלה ושאלה היא למצוא את מספר האפשרויות לסדר שני "אבני יסוד". בשאלה הראשונה "אבני היסוד" היו לבנה עומדת או שתי לבנים צמודות; בשאלה השניה (סעיף א) "אבני היסוד" היו מספר האפשרויות להושיב בנים ובנות על כיסא אחד או על שני כסאות; ובשאלה השלישית "אבני היסוד" היו מספר הדרכים לתא הראשון או לתא השני.

כפעילות בכיתה בעיצוב תוכנה, לאחר שהתלמידים פתרו את החידות והתקיים דיון בפתרונות המנומקים, יתבקשו התלמידים לנסח את השאלה השקולה ובהתאם לבנות יחידת הספריה לסדרת פיבונצי . ניתן לבקש מהתלמידים לחבר חידות דומות שהפתרון שלהן הוא סדרת פיבונצי. יחידת ספריה זו תשמש את התלמידים כדי לכתוב תוכניות לשלוש השאלות, כאשרבהתאם לכל שאלה, יש להחליף את הודעות הפלט ויש לקבוע את הערכים התחיליים של האיברים n=1 ו- n=2.

הרחבה - למתקדמים:

1. ניתן לראות כי כל השאלות שקולות לבעיה כללית אחרת והיא:

"נתון n מספר שלם. מה המספר האפשרויות השונות לפירוק n לסכום של המספרים 1 ו- 2".

לדוגמה:

* עבור n=1 יש רק דרך אחת
* עבור n=2 יש שתי דרכים 1+1=2 או 2
* עבור n=3 יש שלוש דרכים: 1+1+1 או 1+2 או 2+1. (שימו לב: יש חשיבות לסדר האיברים בפעולת החיבור).
* עבור n=4 יש 5 דרכים: 1+1+1+1 או 1+1+2 או 1+2+1 או 2+1+1 או 2+2.

וכך הלאה.

נוכיח שהשאלות שקולות לשאלה זו:

א. קל להראות כי שאלות 1 ו- 3 הן מקרה פרטי של בעיה זו. באופן כללי עבור n כלשהו, ניתן לחלק את הסכומים לשני סוגים:

* רצף המסתיים ב- 1 כמו: 1+1+1+1 או 1+2+1

כל הרצפים המסתיימים ב-1, בלי לכלול את ה-1 האחרון ברצף, הם רצפים של 1 או של 2 שסכומם הוא n-1. אבל מספר הרצפים של 1 ושל 2 כשסכום כל רצף הוא n ובסיומם יש 1 הוא f(n-1).

* רצף המסתיים ב- 2 כמו: 2+2 או 1+2 .

באופן דומה, מספר הרצפים של 1 ושל 2 אשר מסתכמים ב- n ומסתיימים ב- 2 , בלי לכלול את ה- 2 האחרון, הוא f(n-2).

מכאן, מספר הרצפים של 1 ושל 2 שניתן לסכם כדי לקבל את n , הוא $f(n) = f(n-1)+ f(n-2)$.

ב. ניתן להראות שגם שאלה 2 שקולה לבעיה:

"נתון n מספר שלם. מה המספר האפשרויות השונות לפירוק n+1 לסכום של המספרים 1 ו- 2".

דרך אפשרית להראות את השקילות היא לבצע את תהליך ההושבה של הילדים בשלבים, כאשר בכל שלב מושיבים בת אחת או זו ילדים בן ואחריו בת (כדי שלא ישבו שני בנים אחד ליד השני).

כיוון שבבעיה המקורית במקום האחרון ניתן להושיב בן, הרי להושיב n+1 ילדים.

שימו לב שבכל מקרה בכסא האחרון תשב בת. ולכן קיימת התאמה חד-חד ערכית בין הפתרון של שאלה 2 והשאלה הכללית.

2. תיאור של אלגוריתם דייקסטרה לחישוב האיבר ה- n בסדרת פיבונצי יכול להיות פעילות מתאימה להוראת נושא סיבוכיות. במקום לחשב את כל האיברים עד לאיבר ה- n, לפי הנוסחה:

$F(0)=0$

$F(1)=1$

$F(n) = F(n-1) + F(n-2), n>1$

דייקסטרה הציע ב- 1978 אלגוריתם אחר, כאשר ניתן לחשב:

$F(1)=0$

$F(2)=1$

$F(2n-1) = F(n-1)^2 + F(n)^2$

$F(2n) = (2 F(n-1) + F(n)) F(n)$

כאשר אנו צריכים רק את f(n) ואת f(n-1) כדי לחשב את f(2n) וגם את f(2n-1).

לדוגמה, על פי אלגוריתם זה כדי לחשב את f(1000) יש לחשב רק 22 ערכים.

הסבר והוכחה ניתן למצוא באתרים הבאים:

http://www.mcs.surrey.ac.uk/Personal/R.Knott/Fibonacci/fibFormula.html

http://www.cs.utexas.edu/users/EWD/ewd06xx/EWD654.PDF

אתרים:

ישנם אתרים טובים ורבים בעברית ובאנגלית על סדרת פיבונצ'י, השימושים בסדרת פיבונצ'י ועל פיבונצ'י האיש. נציין מספר אתרים:

* על האיש ניתן לקרוא באתר הבא (בעברית

http://web.macam.ac.il/~kobie/math/fibonacci%20site.htm

* בבניית החידה לחודש דצמבר 2003 השתמשנו באתר נפלא הכולל מספר רב של פעילויות וכתובתו:

http://www.mcs.surrey.ac.uk/Personal/R.Knott/Fibonacci/fib.html

* על השימושים של סדרת פיבונצ'י בחיי היום יום ניתן לקרוא גם בכתבה שהתפרסמה בהבטים בהוראת מדעי המחשב, ספטמבר 1997 בשם "פיבונאצ'י – חייו והסדרה הקרויה על שמו", כתבה שרה לב.

פינת ינואר 2004

החידה שאנו מציגים הפעם היא חידה עתיקה (שתוארה ע"י Brahmagupta שחי בהודו בשנים 598 עד 670 וכתב ספרים רבים במתמטיקה ואסטרונומיה):

אישה הלכה לשוק למכור ביצים. בדרך עבר רוכב על סוס ובשוגג הוא פגע בסל הביצים ושבר אותן. הרוכב שרצה לפצות את האישה פנה ואמר: אמרי לי כמה ביצים היו לך בסל? ענתה האישה: אני לא זוכרת כמה ביצים היו לי בסל, אבל כשהוצאתי מהסל בכל פעם 2 ביצים, נשארה רק ביצה אחת בסל. כך גם קרה כשהוצאתי בכל פעם 3 ביצים, 4 ביצים, 5 ביצים ו- 6 ביצים. אבל כשהוצאתי בכל פעם 7 ביצים מהסל, לא נשארה אף ביצה בסל. אמר לה הרוכב: אני יודע כמה ביצים היו לך בסל.

השאלה היא: **מה המספר המינימלי של הביצים שהיו לה בסל?**

פתרון הבעייה: מה מספר הביצים המינימלי שהיו בסל?

בשלב ראשון, נמצא את המספר הקטן ביותר שמתחלק במחלקים- 2, 3, 4, 5 ו-6 עם שארית 1.

נחשב את המספר הקטן שמחלקים אלו הם גורמיו.

מאחר ו- 2, 3 ו- 5 הם ראשוניים, הם חלק מגורמי המספר המבוקש, הגורמים של 6 כלולים במכפלה (3*2), הגורמים של 4 הם 2*2, לכן נוסיף גורם 2 ונקבל:

$$2*2*3*5 = 60$$

וזהו המספר הקטן ביותר שמתחלק ב- 2, 3, 4, 5 ו- 6 ללא שארית.

בשלב שני ננסה למצוא את המספרים המתחלקים ב- 60 עם שארית 1 וגם מתחלקים ב- 7 ללא שארית, כלומר נפתור את המשוואה:

$$60n+1=7m$$

נחלץ את m:

$$m=\frac{60n+1}{7}=6n+\frac{4}{7}n+\frac{1}{7}=6n+\frac{4n+1}{7}$$

מאחר ו- m הוא מספר טבעי, אנו מחפשים 4n + 1 המתחלק ב- 7, ומכאן סדרת המספרים האפשריים הם: 5, 12, 19, 26, ...

(המספר הראשון בסדרה הוא 5 והקפיצות הן 7).

לדוגמה:

* עבור 5, 4*5+1=21 מתחלק ב-7 ללא שארית
* עבור 12, 4*12+1=49 מתחלק ב-7 ללא שארית
* עבור 19, 4*19+1= 77 מתחלק ב-7 ללא שארית

וכן הלאה

אם נציב מספרים אלו בנוסחה 60n + 1, נקבל את המספרים שמהווים פתרון לשאלה: מה מספר

הביצים האפשרי שהיו לאישה בסל. לדוגמה, מספר הביצים האפשרי הוא:

* $60*5 +1 = 301$
* $60*12+1 = 721$
* $60*19+1 = 1143$

כאשר 301 הוא המספר המינימלי של ביצים.

פעילויות בכיתה עם חידת ינואר 2004

רמת קושי

החידה מתאימה לתלמידים ביסודות 1 וקשורה למושגים: div, mod לולאות, ניתוח בעיה, יעילות.

הפעלה בכיתה:

* תחילת הפעילות בהצגת החידה בכיתה ולבקש מהתלמידים לפתור את החידה ולכתוב תוכנית אשר מציגה את המספר המינימלי של הביצים בסל. רצוי לדרוש מהתלמיד פתרון מנומק שמתייחס לנכונות הפתרון ולהסבר מדוע זהו המספר הקטן ביותר.
* פיתוח אלגוריתם: קיימת אפשרות שחלק מהתלמידים אכן יכתבו תוכנית ללא ניתוח כדי לפתור את החידה ב"כוח המחשוב". הדיון במקרה כזה יתמקד במושגים – ניתוח ויעילות של אלגוריתם.

כדי לפתור את הבעיה, עלינו לבצע את המשימות הבאות:

א. למצוא סדרת מספרים שמתחלקים ב- 2, 3, 4, 5 ו-6 עם שארית 1.

ב. מתוך סדרת המספרים שמצאנו ב- א', יש למצוא את אלו שמתחלקים ב- 7 ללא שארית.

האלגוריתם הפשוט ביותר שניתן לכתוב למשימה א' הוא:

אלגוריתם 1

1 ← number (מספר הביצים)

כל עוד (number לא מתחלק ב- 2, 3, 4, 5, 6 עם שארית 1) או (number מתחלק ב- 7 ללא שארית) בצע

number ← number + 1

הצג number

כדי לייעל את התוכנית, נשתמש במידע שקיים בחידה.

למשל: יודעים כי number מתחלק ב- 2 עם שארית 1, כלומר המספרים הם אי זוגיים ואז אפשר לכתוב את ההוראה לדילוג :

number ← number + 2

באופן דומה אנחנו יודעים שהמספר מתחלק ב- 5 ולכן המספרים יכולים להסתיים ב- 0 או ב- 5 (0, 5, 10, 15,..).

למספרים אלו נוסיף 1 (כי ידוע שיש בחלוקה ב- 5 שארית 1), וכך נקבל את המספרים (1, 6, 11,

16, ...).

אבל כל המספרים המסתיימים ב- 6 הם זוגיים וזה סותר את התנאי הקודם , לכן ניתן לשפר ולקפוץ ב- 10 בכל פעם, כלומר:

$$number \leftarrow number + 10$$

כדי לשפר את האלגוריתם של שלב א' , רצוי לחשב מהם המספרים שמתחלקים ב- 2,3,4,5,6 ללא שארית ולהם נוסיף 1, לדוגמה מספר מתאים הוא:

$$2*3*4*5*6 = 720$$

אבל זה לא מבטיח שזה המספר הקטן ביותר שניתן למצוא, למשל גם 360 מתחלק ב- 2, 3, 4, 5 ו- 6.

כדי למצוא את המספר הקטן ביותר, נחפש את המספר הקטן ביותר שפרוק לגורמיו מכיל את המחלקים בחידה.

בעיה דומה שהתלמידים מכירים היא לכתוב תוכנית המוצאת את הגורמים של מספר נתון.

אולם הפעם הבעיה היא הפוכה, אנו יודעים את הגורמים ומחפשים את המספר.

במספר שאנו מחפשים הגורמים שצריכים להיות הם 2, 3, ו- 5 (משום שהם מספרים ראשוניים), אבל 30 = 2*3*5, והוא אינו מתחלק ב- 4. מאחר ו- 4 ניתן לייצג כ- 2*2, נוסיף גורם נוסף 2, כלומר ננסה 60 = 2*2*3*5 המתחלק ב- 2, 3, 4, 5 ו- 6 ללא שארית. כלומר מצאנו את המספר הקטן ביותר שאלו מחלקיו.

כעת עלינו לבצע את המשימה השניה:

מבין המספרים שהם 60n+1 (הכפולות של 60 +1), יש למצוא את המספר הראשון שמתחלק ב- 7 ללא שארית:

אלגוריתם 2

number ← 61

כל עוד (number לא מתחלק ב- 7 ללא שארית) בצע

number ← number + 60

הצג number

הרחבה - למתקדמים:

לכתוב תוכנית המחשבת ומציגה מבין את 10 המספרים שעונים לתנאי החידה, כלומר 301, 721 ,.....

אלגוריתם 3:

number ← 61

לכל מספר מ- 1 עד 10 בצע

כל עוד (number + 1 לא מתחלק ב- 7 ללא שארית) בצע

$number \leftarrow number + 60$

הצג number

$number \leftarrow number+60$

אתרים:

פתרון חידה זו, שייך למשפחה של משוואות דיופנטיות העוסקות בבעיות שפתרונן כרוך בבניית מערכת משוואות שבהן מספר הנעלמים גדול ממספר המשוואות ולכן יש אינסוף פתרונות.

דיופנטוס (Diophantus) הוא מתמטיקאי הלניסטי שחי באלכסנדריה בין השנים 200-284 לסה"נ (בערך).

ספרו היחיד ששרד בחלקו הוא "אריתמטיקה". הספר עוסק בפתרון משוואות מסוגים שונים. משוואות דיופנטיות הן משוואות שבהן מספר המשוואות קטן ממספר המשתנים ולכן יש להן אינסוף פתרונות (לדוגמא $x^2+y^2=z^2$).

בפתרון חידות דיופנטיות אנו דורשים שהפתרון יהיה מספר שלם ותנאים נוספים הנובעים מאופי השאלה.

האתר של דוד שי מכיל חידות דיופנטיות נוספות ופתרון מפורט. אתר זה מומלץ בחום והוא כולל חידות רבות המחולקות לפי נושאים שונים.

http://www.workjoke.com/puzzles/puzzle1m.htm

אתר באנגלית הכולל תיאור מפורט של פתרון משוואות דיופנטיות

http://mathforum.org/library/drmath/view/61325.html

פינת פברואר 2004

בעיית הגמל והבננות:

במטע בננות הממוקם ליד המדבר צמחו 3000 בננות. בעל המטע רוצה למכור את כל הבננות בשוק הנמצא בעיר המרוחקת 1000 ק"מ מהמטע. לבעל המטע יש גמל אחד בלבד המסוגל לשאת בכל זמן מקסימום 1000 בננות. הגמל כדי להתקיים אוכל בננה אחת לכל ק"מ שהוא צועד (הוא אוכל את הבננה לכל אורך הק"מ שהוא צועד).

השאלה היא: מה המספר הגדול ביותר של בננות שבעל המטע יכול להוביל לשוק?

פתרון הבעייה: מה המספר הגדול ביותר של בננות שבעל המטע יכול להוביל לשוק?

נתחיל לבדוק כיצד הגמל יכול לשאת את הבננות ליעד (השוק).
אם הגמל נושא 1000 בננות למרחק 1000 ק"מ, אזי בהגיעו ליעד לא יוותרו בננות למכירה, משום שהוא אכל בננה אחת לכל ק"מ שהוא צעד, כלומר הוא אכל 1000 בננות בדרך. כמו כן, לא יוותרו לו בננות לבצע את הדרך חזרה כדי לקחת את שאר הבננות.
מכאן שהגמל צריך להשאיר בננות בנקודות מסוימות בדרך, כדי שהוא ישתמש בהן בהמשך. נקטין את הבעיה ונניח שהגמל צריך להעביר מקסימום בננות למרחק 1 ק"מ.
אלגוריתם המתאר את הפתרון האופטימלי הוא:

1. הגמל לוקח 1000 בננות מהמקור
2. הגמל נושא אותן ק"מ אחד ואוכל בננה אחת בדרך
3. הגמל משאיר 998 בננות ולוקח בננה אחת לדרך בחזרה
4. הגמל חוזר על צעדים 1 עד 3
5. הגמל חוזר על צעדים 1 עד 2

סיכום ביניים:
כדי להעביר את כל 3000 הבננות למרחק 1 ק"מ הגמל אכל 5 בננות. בסיום האלגוריתם, בנקודה המרוחקת מרחק של 999 ק"מ מהשוק, יהיו 2995 בננות.
אנו יכולים להמשיך את ביצוע האלגוריתם ולהתקדם עוד ק"מ, כשעכשיו אנו יודעים שמחיר לק"מ הליכה הוא 5 בננות. אולם אם נמשיך בדרך זו, הגמל שוב לא יוכל להגיע ליעד משום שהוא יאכל את כל הבננות לפני שיגיע ליעד ($3000/5 < 1000$).

הפתרון מבוסס על הרעיון כי, עבור כמויות שונות של בננות נתן להקטין את המחיר שיש לשלם על הובלתן (או מספר הבננות שיש להאכיל את הגמל). לדוגמה, לאחר שכמות הבננות הנותרות מגיעה ל- 2000, הגמל יאכל רק 3 בננות לכל ק"מ (משום שכעת עליו לבצע רק את הצעדים 1,2,3 ו- 5 באלגוריתם).

מכאן שכדי לפתור את הבעיה, עלינו לחלק את הדרך לקטעים, כאשר בכל קטע המחיר המינימלי להעברת מקסימום בננות הוא קבוע:

* בקטע הראשון - כדי להעביר כמות בננות בין 2001- 3000 בננות למרחק N_1 ק"מ (נקודה A), הגמל צריך לאכול לפחות 5 בננות לכל ק"מ.
* בקטע השני - כדי להעביר כמות בננות בין 1001- 2000 למרחק N_2 ק"מ (נקודה B), הגמל צריך לאכול לפחות 3 בננות לכל ק"מ .
* בקטע השלישי - כדי להעביר כמות בננות בין 1001- 0 למרחק N_3 ק"מ, הגמל צריך לאכול לפחות 1 בננות לכל ק"מ.

כדי להעביר מקסימום בננות ליעד עלינו לחשב את N_1, N_2 ו- N_3 המתאימים:

א. חישוב N_1

$$3000 - 5*N_1 = 2000 \Rightarrow N_1 = 200$$

הנקודה A מרוחקת 200 ק"מ מהמקור.

ב. חישוב N_2

באופן דומה נחשב את המרחק בין A ל- B, הפעם הגמל אוכל 3 בננות לכל ק"מ. כלומר

$$2000 - 3*N_2 = 1000 \Rightarrow N_2 = 333.333 \text{ ק"מ}$$

ג. חישוב N_3

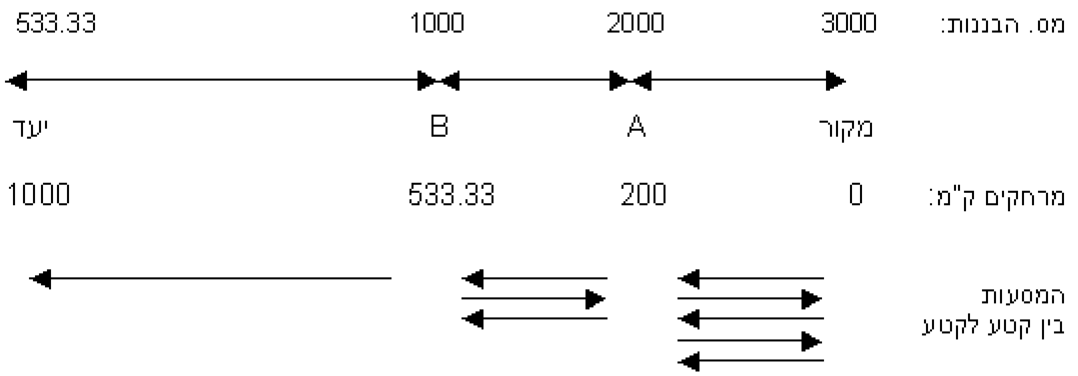
בשלב האחרון בין A ל- B מוביל הגמל 1000 בננות למרחק שנותר והוא:

$$1000 - 200 - 333.333 = 466.667 \text{ ק"מ}$$

כעת לכל ק"מ שנותר (466.67) אוכל הגמל רק בננה אחת ולכן ליעד הוא יוביל:

$$1000 - 466.67 = 533.333 \text{ בננות}$$

לסיכום תרשים גרפי של הפתרון:



פעילויות בכיתה עם חידת פברואר 2004

רמת קושי

חידה ברמה גבוהה, והדגש בה הוא על חקר וניתוח הבעיה.

החידה יכולה לשמש כמבוא ליסודות 1, בשלב בו מוצג המושג אלגוריתם.

הפעלה בכיתה:

* בתחילת הפעילות נציג את החידה בכיתה ונבקש מהתלמידים לפתור את החידה.
* כדי לתמוך בשלב הניתוח מומלץ לבדוק מספר מקרים, לדוגמא:

א. הגמל ישא 1000 בננות למרחק 1000 ק"מ

ב. הגמל יישא את כל הבננות למרחק 400 ק"מ ישאיר 200 בננות ...ואחר יתקדם לעיר

ג. הגמל ישא את כל בננות לקטעים שכל אחד מהם הוא 1 ק"מ (חסר המשך)

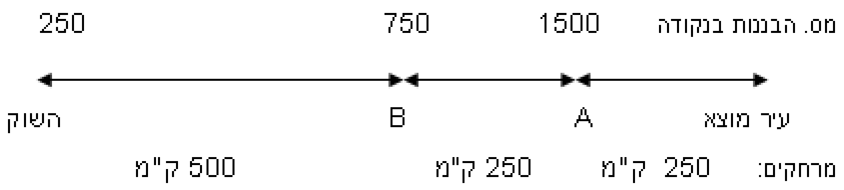
ד. הגמל ישא את כל הבננות למרחקים הבאים: 250 ק"מ, 250 ק"מ ו- 500 ק"מ

בכל אחד מהמקרים הנ"ל יש להציג אלגוריתם מתאים ואת מספר הבננות שהובלו לשוק ואת המחיר ששלם הגמל בכל קטע של הדרך.

לדוגמה עבור מקרה ד, ניתן לכתוב את האלגוריתם הבא:

1. הגמל לוקח 1000 בננות מעיר המוצא
2. הגמל נושא אותן למרחק 250 ק"מ (נקודה A) ואוכל 250 בננות בדרך
3. הגמל משאיר 500 בננות ולוקח 250 בננות לדרך בחזרה
4. הגמל חוזר על צעדים 1 – 3 עד שכל הבננות (חוץ מאלה שאכל) מועברות מהמוצא לנקודה A – בנקודה A יהיו 1750 בננות
5. הגמל לוקח 1000 בננות מנקודה A
6. הגמל נושא אותן 250 ק"מ (לנקודה B) ואוכל 250 בננות בדרך
7. הגמל לוקח 250 בננות לדרך בחזרה לנקודה A
8. הגמל לוקח את 500 הבננות (שנותרו) בנקודה A
9. הגמל נושא אותן 250 ק"מ (לנקודה B) ואוכל 250 בננות בדרך
10. הגמל נושא את כל הבננות מנקודה B (750 בננות) ליעד המרוחק 500 ק"מ ואוכל 500 בננות בדרך

לסיכום, לאחר ביצוע האלגוריתם הגמל הגיע לשוק עם 250 בננות.



* הגדרת הפרמטרים בהם תלוי הפתרון:

מספר הבננות שמוביל הגמל מנקודה לנקודה, המרחק בין הנקודות, מספר המסעות הדרושים לכך = מספר הבננות שיאכל בדרך.

* ניסוח הכללים למציאת התחומים עבורם משלם הגמל 5 בננות לכל ק"מ, 3 בננות לכל ק"מ ובננה אחת לכל ק"מ.

הרחבה:

* נשנה את הבעיה ונשאל:

מה המרחק המקסימלי שהגמל יכול להתקדם אליו אם היו לו בתחילת המסע 3000 בננות ועדיין הוא אוכל בננה אחת לכל ק"מ?

פתרון:

למעשה ניתוח ופתרון הבעיה דומים, אלא שהפעם בהגיעו לנקודה B (שמרחקה מתחילת הדרך הוא 533.33 ק"מ יש לו 1000 בננות ולכן הוא יוכל להתקדם עוד 1000 ק"מ. כלומר הגמל יוכל לצעוד למרחק 1533.33 ק"מ לכל היותר.

* הרחבה מעניינת נוספת הוצעה על ידי **איתן ראט** (תודה!!):

מהו המספר המרבי של בננות שיוכל החקלאי להביא לשוק, אם עומדים לרשותו שני גמלים? ושלושה? ובאופן כללי N גמלים?

מובן שניתן להשאיר גמל/ים בדרך, ולהמשיך רק עם חלקם.

לפתרון של בעיה זו נשתמש בפתרון של הבעיה המקורית: כלומר נחלק את הדרך לקטעים, כשבכל קטע מספר הבננות שאוכלים הגמלים הוא מינימלי כדי להעביר מספר מירבי של בננות לנקודה כלשהי.

* עם שני גמלים, מחיר הובלת הבננות בקטע הראשון יכול להיות 4n (לעומת 5n כשהיה גמל אחד),

ואז החישוב הוא 3000 – 4n = 2000, כלומר n=250km.

מנקודה זו מחיר הובלת 2000 הבננות יורד ל- 2n לכל ק"מ ולכן: 2000 – 2n = 1000,

כלומר נקודה שניה היא במרחק 500 ק"מ מנקודה קודמת או 750 ק"מ מהמוצא.

בקטע האחרון נשתמש בגמל אחד כדי להוביל את 1000 הבננות שנותרו ולכן נשלם 1n לכל ק"מ,

ומאחר שנותרו רק 250 ק"מ, גמל אחד יוביל 750 בננות ליעד.

* באופן דומה נחשב הובלה של בננות עם 3 גמלים ונקבל :

עד נקודה 1: $3000 – 3n1 = 2000 \Rightarrow n1 = 333.33km$

מרחק נקודה 1 מנקודה 2: $2000 – 2n2 = 1000 \Rightarrow n2 = 500$ km

עד היעד נשאר ללכת: $(1000 – n1 – n2) \Rightarrow 167.37$ km

סה"כ בננות שהובלו ליעד: $n1+n2 \Rightarrow 833.33$ bananas

עבור 4 גמלים (או יותר): נשתמש רק ב- 3 גמלים שיכולים לקחת את **כל הבננות** מנקודת המוצא

ולכן אין צורך להשתמש בגמלים נוספים שצורכים דלק (בננות)

מנקודה 1 לנקודה 2, נשתמש רק בשני גמלים כלומר:

(Y מרחק נקודה 2 מנקודה 1) $2000 - 2Y = 1000 \Rightarrow Y= 500km$

סה"כ בננות שהובלו לשוק: $X + Y = 1000/N + 500$.

*** ניתן לשנות את הגרסה הקודמת ולשאול כמה גמלים צריך כדי להוביל מקסימום בננות מהמוצא ליעד.**

מפתרון סעיף קודם ניתן לראות כי עם 3 גמלים נקבל את התוצאה הטובה ביותר.

קישורים:

החידה, הצעה לפתרון חידה וחידות דומות באתר ששמו The Math Forum וכתובתו:

http://mathforum.org/dr.math/faq/faq.camel.html

פינת מרץ 2004

בעיית פלינדרום 1:

עלינו למצוא שלושה מספרים שכל אחד מהם הוא פלינדרום.

המספר הראשון הוא פלינדרום בן 2 ספרות והמספר השני הוא פלינדרום בן 3 ספרות.

מחיבור של המספר הראשון והשני נקבל את המספר השלישי שהוא פלינדרום בן 4 ספרות.

השאלה היא: מהם שלושת המספרים?

בעיית פלינדרום 2:

אחת האפשרויות ליצירת מספר שהוא פלינדרום הוא ביצוע התהליך הבא:

אם נבצע מספר פעמים את הפעולה הבאה: "נחבר למספר טבעי את היפוכו" נקבל מספר שהוא פלינדרום.

לדוגמה: ניקח מספר 461 ונחבר אליו את היפוכו, לדוגמה:

461 + 164 = 625

נחזור על התהליך:

625 + 526 = 1151

1151 + 1511 = 2662

קבלנו פלינדרום.

כדי לבדוק את התופעה לקחנו מספר אחר בן 3 ספרות שאינו פלינדרום, הפכנו אותו וקבלנו מספר שהוא גדול ממנו.

חברנו את 2 המספרים והמספר שהתקבל לא היה פלינדרום.

חזרנו על התהליך ושוב קבלנו מספר חדש בן שלוש ספרות שאינו פלינדרום.

בפעם השלישית שחזרנו על התהליך, המספר שהתקבל היה בן 4 ספרות שאינו פלינדרום,

ואילו בפעם הרביעית שחזרנו על התהליך התקבל מספר בן 4 ספרות שהוא פלינדרום.

השאלה היא: מהו המספר המקורי שאתו התחלנו את התהליך ?

פתרון הבעייה הראשונה: מהם שלושת המספרים הפלינדרומים?

תשובה:

22+979 = 1001

דרך פתרון:

נסמן את המספר הדו-ספרתי כ- aa

ואת המספר התלת ספרתי כ- bcb

לפי תיאור הבעיה:

$$\begin{array}{r} bcb \\ +\quad aa \\ \hline deed \end{array}$$

נקבל

כדי לקבל מספר ארבע-ספרתי מחיבור מספר תלת-ספרתי עם מספר דו-ספרתי,
ספרת המאות של המספר התלת-ספרתי חייבת להיות 9, כלומר 9c9

מחיבור מספר דו-ספרתי הגדול ביותר האפשרי (99) עם מספר תלת ספרתי הגדול ביותר האפשרי (999) נקבל 1098.

כלומר אנו יודעים כי המספר הארבע-הספרתי שצריך הוא 1001.

נסכם את הממצאים עד כה:

$$\begin{array}{r} 9c9 \\ +\quad aa \\ \hline 1001 \end{array}$$

נקבל

כדי לקבל מספר שמסתיים ב- 1 ממספר שמסתיים ב- 9, ע"י סכום,
צריך לחבר לו מספר שמסתיים ב- 2, ולכן המספר הדו-ספרתי הוא 22.

פתרון הבעייה השנייה: מהו המספר המקורי שהתחיל את התהליך?

$$\begin{array}{r} 192 \\ +\quad 291 \\ \hline 483 \\ +\quad 384 \\ \hline 867 \\ +\quad 768 \\ \hline 1635 \\ +\quad 5361 \\ \hline 6996 \end{array}$$

דרך הפתרון:

המספר ההפוך למספר ההתחלתי A גדול ממנו, ולכן ספרת האחדות במספר ההתחלתי גדול מספרת המאות בו.

כלומר, A חייב להיות לפחות 102.

ידוע גם שאחרי שתי פעולות סיכום המספר שמקבל הוא עדיין תלת ספרתי

$$\begin{array}{r} abc \\ +\quad cba \\ \hline def \\ +\quad fed \\ \hline ghi \end{array}$$

אנו יודעים כי def אינו פלינדרום. ולכן, הספרה d שונה מהספרה f. דבר זה אפשרי רק אם d=f+1 (d יכול להיות גדול באחד מ- f משום ש- b הוא לכל היותר 9).

מאחר ו- abc הוא לפחות 102, אז def הוא לפחות 403, כך ש- d+f הוא לפחות 7.

מאחר ו- ghi הוא עדיין מספר תלת-ספרתי שאינו פלינדרום, i יכול להיות לכל היותר 8, כך ש- d+f יכול להיות לכל היותר 8.

מכיוון ש- d=f+1, d+f יכול להיות רק 7, ומכאן ניתן להסיק כי a=1 ו- c=2, כלומר:

$$\begin{array}{r} 1b2 \\ +\ 2b1 \\ \hline 4e3 \end{array}$$

כדי ליצור את מספר 4e3, b צריך להיות 5,6,7,8 או 9. כמו כן, מחישוב של 4e3 עם 3e4 מתקבל:

$$\begin{array}{r} 4e3 \\ +\ 3e4 \\ \hline 8h7 \end{array}$$

מאחר והספרה הראשונה היא 8, e צריך להיות לפחות 5. לכן יש רק שתי אפשרויות עבור b: עבור b=8 נקבל 8+8=16 ועבור b=9 נקבל 9+9=18. כלומר ניתן להסיק ש- b צריך להיות 9.

מסקנה: המספר ההתחלתי הוא : 192.

פעילויות בכיתה עם חידת מרץ 2004

רמת קושי

חידה ברמה בינונית, והדגש בה הוא על חקר וניתוח הבעיה.

החידות מתאימות לתלמידים הלומדים יסודות 1.

הרחבות:

א. אם נוריד את ההנחה שלפיה מהפיכת המספר המקורי מתקבל מספר גדול ממנו, פתרון נוסף אפשרי הוא 291.

ב. שאלה מתבקשת היא האם זו דרך לייצר פלינדרום? כלומר האם האלגוריתם

* קח מספר X

* כל עוד X אינו פלינדרום בצע:

* הפוך מספר X וקבל מספר Y

* X ← X + Y

דוגמאות נוספות:

87	64	13
1. 87 + 78 = 165 2. 165 + 561 = 726		1. 13 + 31 = **44**

3. 726 + 627 = 1353 4. 1353 + 3531 = **4884**	1. 64 + 46 = 110 2. 110 + 011 = **121**

באתר הבא: http://www.jasondoucette.com/cgi-bin/pal.cgi?13 , ניתן למצוא דוגמאות רבות מפורטות.

ובכן מסתבר שהתשובה היא לא.

כמו כן אין לדעת איזה מספר יניב בסופו של תהליך כזה מספר שהוא פלינדרום ואיזה לא.

באתר הבא, ישנו סיפור נחמד על מישהו שביצע חישובים באמצעות מחשב את התהליך המתואר,

כשהמספר ההתחלתי הוא 196.

לאחר שלוש שנים !!! הוא הפסיק את הניסוי (ללא תוצאה).

http://www.fourmilab.ch/documents/threeyears/threeyears.html

סיפור זה מעלה שאלות כמו:

האם אפשר לדעת מראש מתי התהליך מצליח ומתי לא?

מה נוכל לספר לתלמידינו בשאלה זו?

האם זו בעיה לא פתירה?

פינת אפריל 2004

הפעם קיבצנו שתי חידות שתהליך הפתרון בהן דומה:

החידה הראשונה – מחלקים דואר והחידה השנייה ששלח מאיר פריש – חידת הגילאים.

בהזדמנות זו אנו רוצים להודות לאיתן ראט ומאיר פריש על תרומתם הרבה לפורום ולחומרים שנאספו שחלקם מובאים באתר זה.

בבעיות חודש אפריל, בדומה לחידות דיפוניטיות (ראה חידת חודש ינואר), יש פתרון בדרך כלל יחיד (או מספר קטן של בעיות מוגדר) ופתרונן כרוך בחקר של מספרים שמקיימים תנאים מסוימים המוגדרים בבעיה.

חידת הדוור:

בכפר קטן בו 10 בתים הממוספרים מ- 1 עד 10, מחלק ג´ון הדוור את הדואר ששה ימים בשבוע.

באחד הימים סיפר ג´ון בפאב כי בשבוע שעבר שני בתים בכפר לא קיבלו כלל דואר,

לעומת זאת כל שאר הבתים קיבלו 3 פעמים דברי דואר.

בכל יום באותו שבוע הוא חילק דואר ל- 4 בתים בדיוק.

בכל יום, ג'ון חישב ומצא את סכום מספרי הבתים בהם חילק דואר ומצא כי:

* ביום ראשון סכום מספרי הבתים בהם הוא חילק דואר הוא 18
* ביום שני סכומם היה 12
* ביום שלישי סכומם היה 23
* ביום רביעי סכומם היה 19
* ביום חמישי סכומם היה 32
* וביום שישי סכומם היה 25

השאלה היא: אילו שני בתים לא קיבלו כלל דואר באותו שבוע?

חידת הגילאים: (שלח מאיר פריש)

תלמיד ראה את מורהו מדבר עם 3 אנשים.

לאחר שהמורה סיים את שיחתנו ניגש אליו התלמיד ושאל: "מה הגילאים של האנשים שדיברת איתם?"

"סכום הגילאים שלהם הוא כמספר הבית שלך ומכפלת הגילאים שלהם היא 2450", השיב המורה.

חשב התלמיד מספר רגעים ואמר למורהו:"עדיין איני יודע את גיליהם".

השיב לו המורה:"אני יותר מבוגר מהגדול שבהם".

אהה.....השיב התלמיד, סוף סוף אני יודע את גילך.

השאלה היא: מה גיל המורה?

פתרון חידת הדוור: אילו שני בתים לא קיבלו כלל דואר באותו שבוע?

אילו ג'ון היה מחלק דואר בכל הבתים בכפר, ובכל בית בדיוק 3 פעמים, סכום מספרי הבתים בהם

היה מבקר:

$$(1+2+3+4+5+6+7+8+9+10)\times3=165$$

אולם ג'ון ביקר רק בבתים שסכומם הוא: 18+12+23+19+32+25=129

מכך נוכל להסיק שסכום מספרי שני הבתים בהם לא ביקר הוא:

$$(165-129)/3=12$$

האפשרויות ליצור סכום השווה ל– 12 משני מספרים (בין 1 ל- 10) הוא:

א. 2+10
ב. 3+9
ג. 4+8
ד. 5+7

מתיאור הבעיה, נתון כי ביום שלישי ביקר ג'ון בארבעה בתים שסכומם הוא 12, הסכומים האפשריים במקרה זה הם:

ה. 1+2+3+6
ו. 1+2+4+5

גם ביום שישי הוא ביקר ב- 4 בתים שסכומם הוא 32 שאותו ניתן להרכיב בשתי צורות:

ז. 5+8+9+10
ח. 6+7+9+10

מכאן אנו יכולים להסיק כי בבתים שמספרם: 1, 2, 9 ו- 10 ג'ון בוודאות חילק דואר (לא ניתן להרכיב את הסכום 12 ללא 1 ו- 2 ובאופן דומה לא ניתן להרכיב את הסכום 32 ללא 9 ו- 10).

כמו כן אנו יכולים להסיק שהקומבינציות 2 +10 (א) ו- 3+9 (ב) לא אפשריות.

גם הצרוף 5 +7 (ד) אינו אפשרי משום שהוא ביקר או בבית מס. 5 או בבית מס. 7.

לכן נותר רק הצירוף 4+8 = 12 ובבתים אלו הוא לא ביקר כלל.

פתרון חידת הגילאים: מה גיל המורה? (שלח איתן ראט)

נפרק את 2450 למכפלת גורמיו הראשוניים: 2450 = 2 * 5 * 5 * 7 * 7.

מכיוון שהמורה *דיבר* עם האנשים, ניתן להסיק שהצעיר שבהם לא היה בן שנתיים.

אי לכך, אין כאן הרבה אפשרויות. האפשרויות הן:

* אפשרות א׳: 5, 10, 49. סכום גילים: 64
* אפשרות ב׳: 5, 14, 35. סכום גילים: 54
* אפשרות ג׳: 7, 10, 35. סכום גילים: 52
* אפשרות ד׳: 5, 7, 70. סכום גילים: 82
* אפשרות ה׳: 7, 14, 25. סכום גילים: 46

* אפשרות ו´: 7, 7, 50. סכום גילים: 64

מכיוון שלפי המכפלה ולפי סכום הגילאים הילד עדיין לא ידע את גיל המורה, יוצא שרק אפשרות א´ או ו´ נכונה

(כי סכום הגילאים בהן שווה, וזהו כנראה גם מספר הבית שבו גר הילד).

מאחר ובכך שאמר המורה שהוא מבוגר יותר מהגדול שבהם, ידע הילד את גילו, יוצא שגילו של המורה הוא בדיוק 50.

אילו היה יותר מבוגר, אז היה יותר מבוגר מהגדול שבין האנשים בשתי האפשרויות, והילד עדיין לא היה יודע את התשובה.

אילו היה צעיר יותר, לא היה מבוגר מאף אחד מבין הגדולים.

פעילויות בכיתה עם חידת הגילאים - אפריל 2004

רמת קושי

חידה ברמה בינונית, והדגש בה הוא על חקר וניתוח הבעיה.

החידות מתאימות לתלמידים הלומדים יסודות 1 ו-2.

אנו משתמשים בפעולות div ו- mod, פעולות חשבון בסיסיות, פעולות תנאי ולולאות.

הרחבות:

לחידת הגילאים הציע מאיר פריש את ההרחבה הבאה:

האם קיימת מכפלת גילאים נוספת (מעבר ל- 2450) הנותנת גם פיתרון יחיד לבעיה הנ"ל?

(רק לאחר הרמז שנתן המורה לגבי גילו).

הדרכה: דוגמה מצוינת לכתיבת אלגוריתם ותכנית מחשב שיחפשו עוד מספר כנ"ל.

כפתרון להרחבה נציע שתי תכניות:

* תכנית של איתן ראט בשפת פסקל בשם ages.pas
* תכנית של שרה פולק בשפת פרולוג בשם ages.pro

פעילויות בכיתה עם חידת הגילאים - אפריל 2004
תכנית בשפת פסקל

* תכנית של איתן ראט בשפת פסקל בשם ages.pas

```
Program Teacher_and_People;
Uses Crt;
Const
    N = 3; { Number of requested multipliers }
    Max_Sols = 500;
Type
```

```
    TDigit       = 0..9;
    TMultipliers = Array[1..N] of Integer;
    TAll_Mult    = Array[1..Max_Sols] Of TMultipliers;
    TSums        = Array[1..Max_Sols] Of Integer;

Var
    Num   : Integer;
    All   : TAll_Mult;
    Found : Integer;

{=============================================================}
    Procedure Find_All_Mult (Num       : Integer;
                             N         : Integer;
                             Var All   : TAll_Mult;
                             Left      : Integer;
                             Var Found : Integer);
    Var
        Temp, I : Integer;
    Begin
        If (Left = 1) And (Num > All[Found+1, N-Left]) Then
          Begin
              Inc(Found);
              All[Found,N] := Num;
              For I := 1 To N Do
                 All[Found+1,I] := All[Found,I];
          End
        Else
           If (Left > 1) Then
           Begin
               If (Left = N) Then
                  Temp := 1
               Else
                  Temp := All[Found+1,N-Left];

               For I := Temp To Num Do
                 Begin
                     If (Num mod I = 0) Then
                       Begin
                           All[Found+1,N-Left+1] := I;
                           Find_All_Mult(Num div I, N, All, Left-1, Found);
                       End;
                 End;
           End;
    End;

{=============================================================}

    Procedure Delete_Those_With_Wrong_Sum(Var All : TAll_Mult);
    Var
        I, J, K, Min, Max, Temp, MultSum : Integer;
```

```
    Sums : Array[1..Max_Sols] Of Integer;
Begin
    { Calc the sum of each solution }
    For I := 1 To Found Do
       Begin
           MultSum := 0;
           For J := 1 To N Do
              Inc(MultSum, All[I,J]);
           Sums[I] := MultSum;
       End;

    { Delete the solutions with unique sum }
    I := 1;
    While (I <= Found) Do
       Begin
           J := 1;
           While (J <= Found) And ((Sums[J] <> Sums[I]) Or (J = I)) do
                Inc(J);
           If (J = Found+1) Then
             Begin
                 For J := I+1 To Found Do
                    Begin
                        All[J-1] := All[J];
                        Sums[J-1] := Sums[J];
                    End;
                 Dec(Found);
             End
           Else
              Inc(I);
       End;

    { Delete the solutions that the difference between the biggest
      numbers in them is not one. }
    I := 1;
    While (I < Found) Do
       Begin
           { Find the minimum and maximum of the biggest multiplier
             of the combinations that give the sum of the i-th place}
           Min := All[I,N];
           J := 1;
           While (J <= Found) and ((Sums[J] <> Sums[I]) Or (I = J)) Do
                Inc(J);
           Max := All[J,N];
           If (Min > Max) Then
             Begin
                 Temp := Min;
                 Min  := Max;
                 Max  := Temp;
             End;
           For J := J+1 To Found Do
```

```
                  If (Sums[J] = Sums[I]) Then
                     If (All[J,N] > Max) Then
                       Max := All[J,N]
                     Else
                        If (All[J,N] < Min) Then
                          Min := All[J,N];

               { If the difference betwin Min and Max is not 1, then
                 these combinations are not what we want - delete them.}
               If (Max - Min <> 1) Then
                  Begin
                      Temp := Sums[I];
                      J := I;
                      While (J <= Found) Do
                         If (Sums[J] = Temp) Then
                            Begin
                                For K := J+1 To Found Do
                                   Begin
                                       All[K-1] := All[K];
                                       Sums[K-1] := Sums[K];
                                   End;
                                Dec(Found);
                            End
                         Else
                            Inc(J);
                  End
               Else
                  Inc(I);
           End;
    End;
    {========================================================}

    Procedure Print (Num : Integer; All : TAll_Mult; Found : Integer);
    Var
        I, J : Integer;
    Begin
        For I := 1 To Found Do
           Begin
               Write(Num,' = ');
               For J := 1 To N-1 Do
                  Write(All[I,J]:2,' * ');
               Writeln(All[I,N]);
           End;
    End;

Begin
    ClrScr;
    For Num := 3 to 1000 Do
       Begin
           Found := 0;
```

```
        Find_All_Mult(Num, N, All, N, Found);
        Delete_Those_With_Wrong_Sum(All);
        If (Found > 1) Then
          Begin
              Print(Num, All, Found);
              {ReadKey;}
          End;
    End;
End.
```

פעילויות בכיתה עם חידת הגילאים - אפריל 2004

תכנית בשפת פרולוג

```
גילים([2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25]).

מכפלת_גילים(X, A, B, C):-
    גילים(_רשימה),
    חבר(A, _רשימה),
    חבר(B, _רשימה), A \= B,
    חבר(C, _רשימה), C \= B, A\=C,
    X הוא A*B*C,
    X > 2450,
    X < 3000.

סכום_גילים(_גיל, [_א, _ב, _ג], [_א1, _ב1, _ג1]):-
    מכפלת_גילים(_גיל, _א, _ב, _ג),
    מכסימילי([_א, _ב, _ג], _מ),
    _א1 הוא _מ +1,
    מכפלת_גילים(_גיל, _א1, _ב1, _ג1),
    מכסימילי([_א1, _ב1, _ג1], _א1),
    לא (תמורה([_א, _ב, _ג], [_א1, _ב1, _ג1])).

תמורה([],[]):- !.
תמורה([_א|_ר1], _ר2):- חבר(_א, _ר2), תמורה(_ר1, _ר2).

מכסימילי(L, _מ):- חבר(_מ, L), לא (חבר(_אחר, L), _אחר > _מ).
```

פינת מאי 2004

לסיום הפעילות לשנת תשס"ד נציג משחקים כמקור נוסף לבעיות.

נציג שני משחקים כדוגמה למשחקים שבהם ניתן להשתמש ביסודות 1-2.

"החידה" שעלינו לפתור במשחק היא בעצם בחירת אסטרטגית משחק שתבטיח ניצחון.

בהמשך, תוכלו למצוא את תאור המשחקים, הפעילויות בכיתה והניתוח של אסטרטגיות המשחק בכפיפה אחת.

משחק 1:

המשחק הוא לשני שחקנים. נתונה ערימה עם X גפרורים ($X \geq 1$). כל שחקן בתורו יכול לקחת בין 1 ל- 3 גפרורים. מי שלוקח את הגפרור האחרון מפסיד.

לדוגמה, אם בערימה יש 4 גפרורים, כדי לנצח השחקן שתורו לשחק ייקח 3 גפרורים, וכך יותיר בערימה גפרור אחד. מהלך זה יאלץ את היריב במהלך הבא לקחת את הגפרור שנותר ולהפסיד במשחק.

המטרה: למצוא אסטרטגיה לניצחון ולכתוב תכנית מחשב של משחק אדם נגד המחשב.

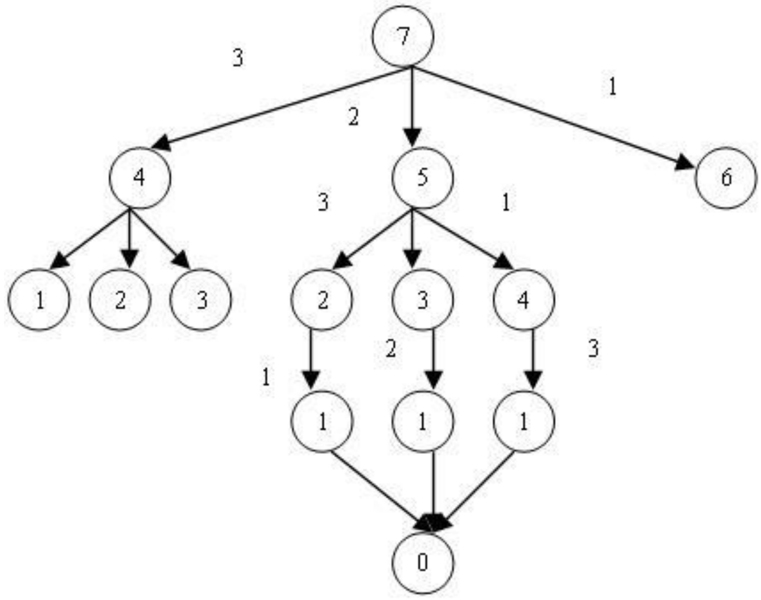
תחילה נציג את האסטרטגיה המתאימה ואחר כך נתאר את המרכיבים של תכנית מתאימה.

בשלב ראשון מומלץ שהתלמידים ישחקו בזוגות מספר פעמים וירשמו לעצמם פעולות מוצלחות. לאחר מכן, נקיים דיון בכיתה בשאלה: מהי האסטרטגיה המומלצת לשחק במשחק זה כדי לנצח.

כדי להדגים מצבים שונים במשחק ניתן להשתמש בעץ הפורש את האפשרויות השונות החל ממצב נתון.

נתאר לדוגמה, עץ המתאר באופן חלקי מצבים אפשריים במשחק בו הערימה ההתחלתית היא בת 7 גפרורים. השורש בעץ מכיל את המצב ההתחלתי ובכל רמה אנו פורשים חלק מהמהלכים האפשריים ואת המצבים שיווצרו לאחר הפעלתם.

בעץ המתואר באיור הבא, הרמה הראשונה (רמת אפס היא השורש) מתארת את הפעולות שיכול שחקן א' לבצע (מידע רשום על הקשתות) ואת המצבים האפשריים של ערימת הגפרורים לאחר ביצוע הפעולות. הרמה השנייה בהתאמה פורשת את הפעולות והמצבים של שחקן ב'. על הקשתות רשומים מספר הגפרורים שנלקחו. כך נוכל להמשיך ולפרוש את העץ, רמה אחת לשחקן א' ורמה הבאה לשחקן ב'.



עץ כזה יכול לסייע לנו למצוא מצבים שיובילו לניצחון.

לדוגמה, על פי האיור, ניתן לראות שאם שחקן ייקח מהערימה בת 7 גפרורים 2 גפרורים, בערימה יישארו 5 גפרורים ומצב זה יאפשר לו בסופו של דבר ניצחון. מערימה של 5 גפרורים, היריב יוכל לבחור בין גפרור אחד לשלושה, כך שמספר הגפרורים יקטן ויהיה 2 או 3 או 4 (בהתאמה). כעת על השחקן הראשון לקחת מספר גפרורים כך שהערימה שהיריב יקבל, יישאר בדיוק גפרור אחד, אותו הוא יאלץ לקחת וכך להפסיד.

הבעיה היא שבדרך כלל עצים כאלו הם גדולים מאוד וקשה לשרטט את כל המצבים האפשריים במשחק. אולם מאחר ומשחק זה הוא פשוט, קל לעקוב ממצבים שבהם המשחק הסתיים, דהיינו מהעלים של העץ, ולבחון את המסלול שהוביל לניצחון, ובהתאם לנסות להכליל אסטרטגיה מתאימה.

באופן כללי במשחק זה האסטרטגיה שמבטיחה ניצחון, היא לקחת Y גפרורים מערימה בה X גפרורים, כך שמספר הגפרורים שיישאר R יקיים:

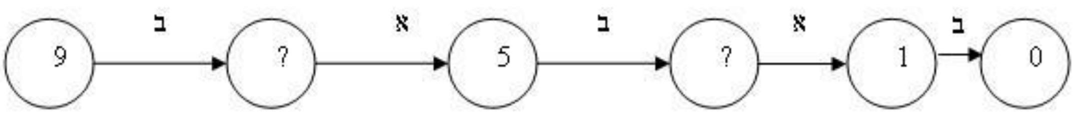
$$R \bmod 4 == 1$$

נחזור לדוגמה בעץ.

ממצב בו היו $X = 7$ גפרורים, לקחנו $Y = 2$ כך שמספר הגפרורים שנשארו הוא $R = 5$,

כאשר מתקיים: $5 \bmod 4 == 1$.

ממצב זה, כפי שתיארנו קודם לכן, לא חשוב מה מספר הגפרורים שיריב ייקח (1,2 או 3 גפרורים), אנו נוכל להוביל מהמצב החדש שיתקבל (2, 3 או 4 גפרורים בערימה) למצב שבו יישאר גפרור אחד הערימה אותו היריב יאלץ לקחת. באופן כללי המסלול הבא מוביל לניצחון של שחקן א:



לסיכום, אם מספר הגפרורים בערימה התחלתית אינו מקיים את התנאי, כלומר $X \bmod 4 \neq 1$,

השחקן הראשון (שמתחיל במשחק) יכול להבטיח לעצמו ניצחון אם הוא ייקח Y גפרורים
כך שמספר הגפרורים שיוותרו יקיים: x mod 4 ==1.
אולם, אם מספר הגפרורים בערימה ההתחלתית מקיים את התנאי x mod 4 ==1,
השחקן השני ינצח אם הוא ישחק על פי אותה אסטרטגיה.

כתיבת תכנית מחשב במשחק זה היא פשוטה יחסית משום שלא דרוש גרפיקה מיוחדת, מבנה נתונים פשוט כולל משתנה מטיפוס שלם עבור ערימת הגפרורים והאלגוריתם ניתן ליישום במסגרת יסודות 2:

1. הגרל מספר גפרורים X
2. קבע שחקן מתחיל
3. כל עוד יש גפרורים בערימה בצע:
    - 3.1 אם תור שחקן (שאינו מחשב) אזי
        - 3.1.1 קלוט מספר גפרורים (חוקי) שלוקח השחקן
        - 3.1.2 עדכן ערימה
    - 3.2 אחרת (תור מחשב)
        - 3.2.1 חשב Y מספר גפרורים שיש לקחת כך ש: (X – Y) mod 4 == 1
        - 3.2.2 הודע Y גפרורים נלקחו
        - 3.2.3 עדכן ערימה
    - 3.3 הצג ערימה מעודכנת
    - 3.4 אם יש ניצחון (מספר גפרורים 0) אזי
        - 3.4.1 הודע על המנצח
    - 3.5 אחרת החלף שחקן

משחק 2:

שוב משחק לשני שחקנים.
נתון לוח שהוא שורה של 9 ריבועים ממוספרים מ- 1 עד 9. השחקנים בתורם בוחרים אחד הריבועים.
המנצח הוא השחקן הראשון שבחר 3 מספרים שסכומם הוא 15. המטרה היא לכתוב תוכנית של משחק בין מחשב לאדם.

בשלב ראשון נאפשר לתלמידים להתנסות במשחק בזוגות.
בשלב השני נדון ברעיונות לתכנית המתארת את מהלכי המחשב.
כדי לנצח במשחק, יש לבחור שלושה של מספרים שסכומם 15. עם זאת הפתרון הבעיה כתכנית אינו טריוויאלי

משום שעלינו למצוא מבנה נתונים מתאים שבעזרתו נוכל לחשב את השלשות באפשריות ומהן לנסות לבחור בשלשה מסוימת שתוביל לניצחון. בנוסף עלינו לדאוג שביריב לא ינצח ובמהלך המשחק למנוע ממנו לבחור בשלשה שתביא אותו לניצחון.

לדוגמה, נתאר מהלכים אפשריים בעזרת הטבלה הבאה:

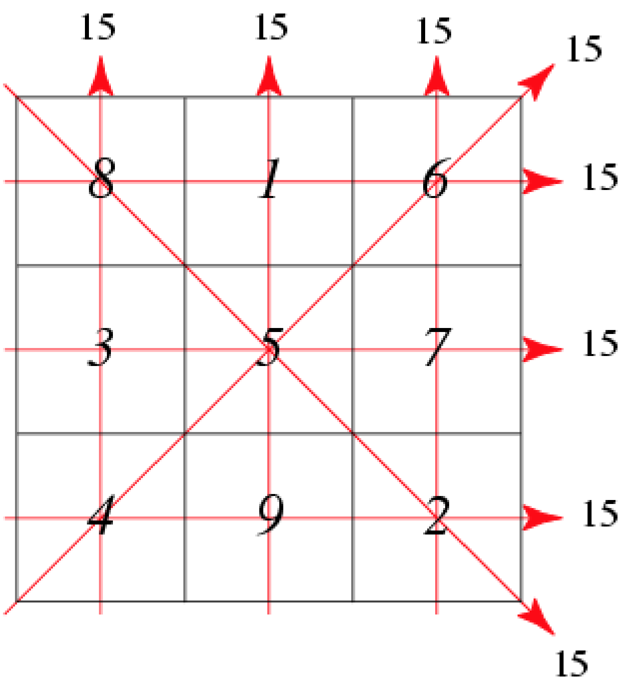
מס. מהלך	שחקן א	יריב	
1		9	
2	4		
3		5	
4	1		שחקן א בחר בערך 1 כדי למנוע מהיריב לקבל שלשה {9, 5, 1} שסכומם 15
5		8	
6	7		
7		2 (ניצחון)	

המשחק נראה תחילה פשוט אך ככל שמתקדמים, יש יותר אפשרויות לסיכום ולכן הוא הולך ומסתבך.

במשחק זה, הקושי נובע מהצורך לשמור מידע על שלשות של מספרים אפשריים.

ניתן לפתור קושי זה אם נשתמש בריבוע קסם שסכום השורות, העמודות והאלכסונים בו הוא 15.

להלן ריבוע קסם כזה:



סכום שורות

4 + 9 + 2 = 15

3 + 5 + 7 = 15

8 + 1 + 6 = 15

סכום עמודות

4 + 3 + 8 = 15

9 + 5 + 1 = 15

2 + 7 + 6 = 15

סכום אלכסונים

4 + 5 + 6 = 15

2 + 5 + 8 = 15

משחק זה דומה למשחק המפורסם TicTacToe או איקס-עיגול והוא מסתיים בדרך כלל בתיקו אלא כן אחד השחקנים ביצע טעות. במשחק זה למחשב יש יתרון משום שמספר המצבים (גם אם הוא גדול) הוא סופי. כמו כן למחשב יש לו ידע על כל האפשרויות שעומדות בפני היריב. מידע זה והמהירות שבו המחשב יכול לאתר את השלשות האפשריות (שים לב אין צורך לחשב את הסכום אלא רק לדעת את צרוף השלשות) מקנה לו יתרון על היריב.

בכתיבת התכנית אנו נבסס את אסטרטגית המשחק של המחשב על ביצוע אחת משתי הפעולות הבאות:

* לבדוק שהיריב אינו עומד לנצח בשלב הבא, אם כן יש לחסום אותו על ידי בחירת המספר הבא שהיריב מתכנן לבחור כדי לנצח.
* לבחור מספר שמאפשר הרכבה של שלשות רבות. טקטיקה זו מאפשרת למחשב להתאושש במהירות כאשר היריב חוסם אותו. במקרה כזה המחשב יכול להרכיב שלשה אחרת (שחלק ממספריה הם המספרים שהוא בחר עד כה). מסיבה זו בתכנית המחשב נעדיף לבחור בערך 5 שהוא בריבוע האמצעי ואם זה אינו אפשרי, נעדיף לבחור באחד המספרים שבפינות של ריבוע הקסם (4, 2, 8, 6).

הרחבה:

ניתן להתייחס לריבועי קסם נוספים שסכומם אינו 15 או ריבוע קסם שמכילים nxn שורות ועמודות.

כמובן שאם סכום מסוים יש אפשרות ליצור יותר מריבוע קסם אחד, רמת הקושי של המשחק עולה.

כתיבת תכנית במקרה זה היא מורכת יותר ודורשת תכנון של מבנה נתונים מורכב יותר – טבלה דו ממדית המכילה את ריבוע הקסם האפשרי וטבלה נוספת באותו גודל המכילה הצבעה בכל תא, אם המספר נבחר ואם כן על ידי מי.

לדוגמה, ניתן לציין 0 למספר שעדיין לא נבחר, 1 לציון מספר שנבחר על ידי היריב ו- 2 לציון מספר שנבחר על ידי המחשב. (ניתן להשתמש בטבלה אחת המאותחלת כריבוע קסם, וסימון 0 או 1- לציון השחקן שבחר במספר מסוים).

כהרחבה, בסיום הפעילות הזו, תלמידים יכולים לכתוב תכנית לניהול משחק איקס עיגול. הקישורים בהמשך כוללים הפניה למשחקים מסוג זה.

קישורים:

** המשחק לקוח מאתר בשם: cut the knot שכתובתו:

http://www.cut-the-knot.org/games.shtml

האתר מכיל בעיות מתמטיות רבות מסוגים שונים והוא מומלץ מאוד.

** שני אתרים על ריבועי קסם

http://mathworld.wolfram.com/MagicSquare.html

http://mathforum.org/alejandre/magic.square.html

** הצעות לכתיבת תכנית למימוש איקס-עיגול

http://en.wikipedia.org/wiki/Tic-tac-toe

http://homepages.cwi.nl/~rooij/tictactoe/strategy.html