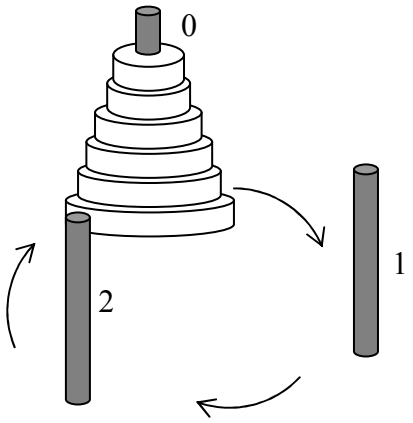
שאלה 2 (20 נקודות)

שאלה זו היא וריאציה על בעיית מגדלי הנוי.

נתונים שלושה מוטות מסודרים על מעגל. על אחד מהם מושחלות hn טבעות בגדלים שונים בערמה, כך שמעל כל טבעת (מלבד העליונה ביותר) שוכבת טבעת קטנה ממנה. כמו בבעיית מגדלי הנוי המקורית, בכל שלב לא תונח טבעת על טבעת קטנה ממנה. בנוסף לכך, העברת הטבעות נעשית רק בין מוטות סמוכים **עם כיוון השעון בלבד.** כלומר, מותר להעביר טבעת רק ממגדל 0 למגדל 1, ממגדל 1 למגדל 2, וממגדל 2 למגדל 0.
אין להעביר טבעת ישירות ממגדל 0 למגדל 2.



כתבו פונקציה **רקורסיבית** בשם CyclicHanoi שמקבלת כפרמטר את מספר הטבעות n, את מגדל המקור source, ומגדל היעד target כמספרים שלמים (0, 1, או 2), ומדפיסה את הפתרון לבעיה. ניתן להניח כי source ≠ target. ניתן להשתמש בפונקציה move אשר מדפיסה את ההודעה המתאימה להעברת טבעת ממגדל source למגדל target אך ורק אם ההעברה היא בכיוון השעון. חתימת הפונקציה move מוגדרת כ:

```
void move(unsigned int source, unsigned int target);
```

אנא ממשו את הפונקציה CyclicHanoi בדף הבא.

```
void CyclicHanoi(unsigned int n, unsigned int source, unsigned int target) {

  unsigned int aux = 3-source-target;

  if (n == 0)
    return;

  if (next(source) == target)
  {
    CyclicHanoi(n-1,source, aux);
    move(source, target);
    CyclicHanoi(n-1, aux, target);
  }
  else {
    CyclicHanoi(n-1, source, target);
    move(source, aux);
    CyclicHanoi(n-1, target, source);
    move(aux, target);
    CyclicHanoi(n-1, source, target);
  }
}

int next(int polenum){
  return (polenum+1)%3;
}
```