שאלה 4 (25 נקודות)

בשאלה זו נרצה לפתור את בעיית **מגדלי הוואי**. בדומה לבעיית מגדלי הנוי, בבעיה יש שלושה מגדלים (נסמנם 1, 2 ו-3), ומספר טבעות המונחות על אחד המגדלים. ואולם כיאה להוואי, ובניגוד לבעיית מגדלי הנוי, הטבעות הן צבעוניות: מכל גודל טבעת יש שתי טבעות, האחת אדומה והשנייה לבנה, ובסה"כ יש m טבעות אדומות ו-m טבעות לבנות.

במצב ההתחלתי הטבעות ממוקמות זו מעל זו במגדל מס' 1 ומסודרות לפי גודלן (הגדולה ביותר למטה). מכל זוג טבעות באותו הגודל, זו הלבנה ממוקמת מתחת לזו האדומה. למשל, עבור m=3 המצב ההתחלתי הוא:

כללי הבעיה נותרים כמו בבעיה המקורית:

- אין להניח טבעת גדולה מעל טבעת קטנה ממנה. עם זאת, **מותר** להניח שתי טבעות מאותו הגודל האחת מעל לשנייה, ללא קשר לצבע (כלומר מותר לבנה על אדומה וגם אדומה על לבנה מאותו הגודל).
- אין להזיז יותר מטבעת אחת בו זמנית.

בשאלה זו אין הגבלה על סיבוכיות הפונקציות. לשם הזזת הטבעות, ניתן להשתמש בפונקציות הבאות:

```
void movered(int from, int to) {
  printf("move red ring from %d to %d\n", from, to);
}
```

```
void movewhite(int from, int to) {
  printf("move white ring from %d to %d\n", from, to);
}
```

סעיף א

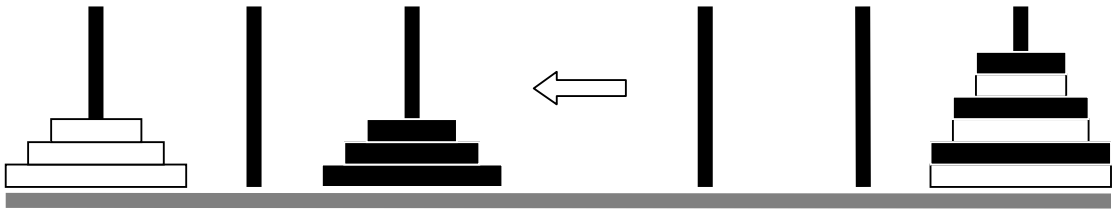
כתבו פונקציה שמקבלת את m – מספר זוגות הטבעות במגדל המקור (כלומר סה"כ ישנן 2m טבעות, שתיים בכל גודל), וכן את מספר מגדל המקור ומספר מגדל היעד. הפונקציה מדפיסה את ההוראות להעברת כל הטבעות ממגדל המקור למגדל היעד, תוך שימוש במגדל השלישי כמגדל עזר. במצב הסופי, על הטבעות להיות **באותו הסדר** כמו שהיו במגדל המקור – כלומר לבנה מתחת לאדומה. לדוגמה, עבור m=1, מגדל המקור 1 ומגדל היעד 3, הפונקציה תדפיס:

```
move red ring from 1 to 2
move white ring from 1 to 3
move red ring from 2 to 3
```

```
void hawaii1(int m, int from, int to) {
```

סעיף ב

כתבו פונקציה שמקבלת את m – מספר זוגות הטבעות במגדל המקור, וכן את מספר מגדל המקור ומספר מגדל היעד. הפונקציה מדפיסה הוראות להעברת הטבעות, כך שבסוף התהליך הטבעות מופרדות על פי צבען – כל הלבנות נמצאות במגדל היעד (הגדולה למטה) וכל האדומות נמצאות במגדל המקור (הגדולה למטה). המצב ההתחלתי והמצב הסופי נראים כך:



לדוגמה, עבור m=1, מגדל המקור 1 ומגדל היעד 3, הפונקציה תדפיס:

```
move red ring from 1 to 2
move white ring from 1 to 3
move red ring from 2 to 1
```

בסעיף זה ניתן להשתמש בפונקציה מהסעיף הקודם, גם אם לא מימשתם אותה.

```
void hawaii2(int m, int from, int to) {
```