Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра математики факультету інформатики

Курсова робота на тему:

«ЗАДАЧА ПРО ХАНОЙСЬКУ ВЕЖУ ДЛЯ ПОВНИХ БІНАРНИХ ДВОКОЛЬОРОВИХ ДЕРЕВ»

Керівник курсової роботи
Доктор фіз-мат. наук, професор
Олійник Богдана Віталіївна

____

(підпис)

“11” травня 2020 р.

Виконала студентка 1 курсу
напряму підготовки
за науково-освітньою програмою
“Прикладна математика” 113
Сушарник Діана
“11” травня 2020 р.

Зміст

1. Вступ

Уже протягом кількох сторіч [1] задача про Ханойську вежу приваблює інтерес математиків. На сьогоднішній день існує безліч варіацій та узагальнень цієї головоломки: експерементують із кількістю кілків, дисків, розглядають задачі із використанням кольорових елементів, смугасті задачі [2]. Розраховують кількість необхідних кроків і оптимальний алгоритм розв’язку [3]. Але нові варіанти і додаткові умови продовжують з’являтися, тож задача є гарним полем для математичних роздумів.

Під час дослідження цієї теми я розглянула модифікацію задачі про Ханойську вежу для бінарних дерев [4], а також узагальнений розв’язок до модифікованої задачі. У роботі сформульовано правила для нової задачі на бінарних двокольорових деревах і знайдено її розв’язок для дерева фіксованої висоти. Цей напрямок можна продовжувати розвивати і знайти, наприклад, узагальнений розв’язок задачі, або порахувати кількість необхідних кроків для дерев більшої висоти, із використанням іншої кількості кольорів, динамічним розфарбовуванням, тощо.

2. Різновид задачі про Ханойську вежу із використанням бінарних дерев

Загальновідому гру про Ханойську вежу можна узагальнити таким чином, щоб замість стосу дисків переміщувати з одного місця на інше граф.

Так, до прикладу, нідерландський науковець Йост Енгельфріет модифікував цю задачу для повних бінарних дерев. [4]

За його підходом, V-подібні піддерева, що розташовані на різних рівнях бінарного дерева, виконують роль дисків, нанизаних на кілок (рис. 1). За умовами гри, за один хід ми можемо переміщувати одне V-подібне піддерево, якщо воно розташоване на верхівці дерева (тобто його кінцеві вузли є листками).

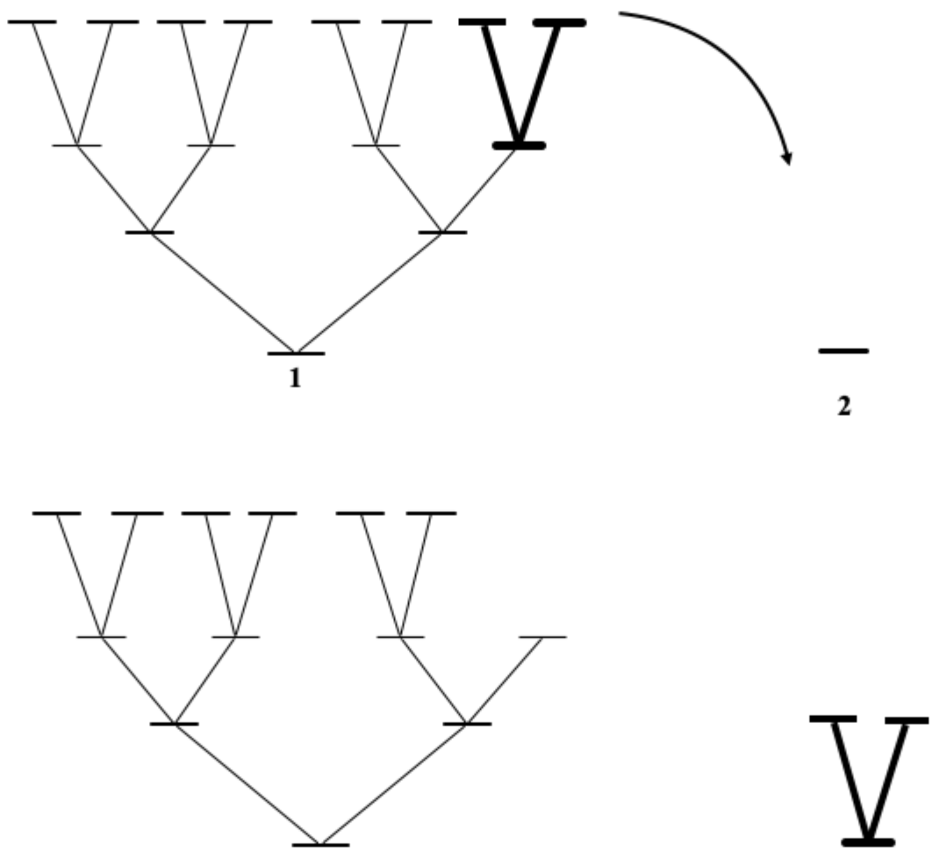


Рис.1 Переміщення одного елементу за один крок

2.1. Початкова конфігурація та принцип гри

За аналогією до гри у Ханойську вежу, потрібно ввести додаткову умову у конфігурацію задачі: нехай, вузли одного рівня однакові між собою, але завжди менші за вузли батьківського рівня (рис. 2).

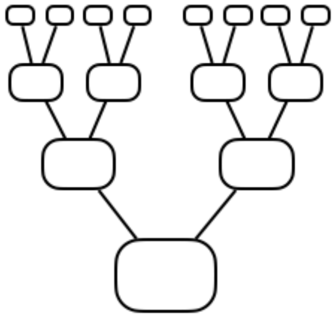


Рис.2

На кожному кроці гри можна переміщувати пару листків [5] одного V-подібного піддерева, тобто вузлів без синів, з одного місця на будь-яке інше вільне місце, де вони знову будуть листками V-подібного піддерева, лише за умови, що вони повиинні бути меншими за свого батька. Кореневий вузол початкового дерева не переміщується. Вважаємо, що будь-яка з «платформ» (вільних місць) є тотожна кореневому вузлу того піддререва, яке стає на нього, і стане головним «коренем», як тільки усі синівські піддерева розмістяться на неї.

Зокрема, можливі випадки, коли листки встають або на ще зовсім вільне місце (приєднуються до «платформи»), або на таке місце, де вони можуть приєднатися до дерева у якості синівського V-подібного піддерева (якщо така позиція вільна). Таким чином, у кожен момент часу гри кожне непорожнє місце містить двійкове дерево (не обов’язково повне), в якому кожен син є меншим за свого батька (за наявності батька).

У своїй задачі Енгельфріет розглядає повне бінарне дерево висоти 3, але на відміну від класичної гри, в якій використовується три основи для дисків, застосовується чотири «платформи» для переміщення об’єктів (рис. 3)

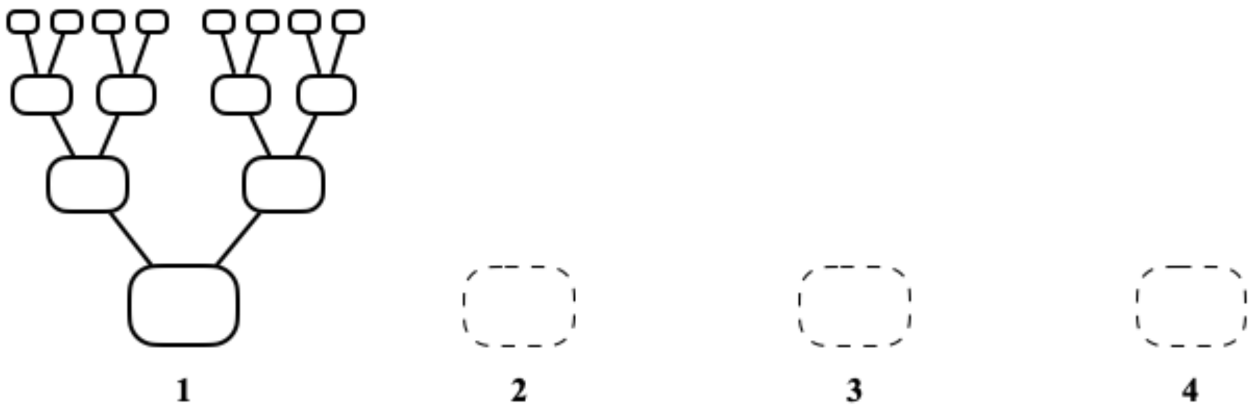


Рис. 3 Чотири платформи для переміщення об’єктів

Кінцевою метою гри є повністю перемістити бінарне дерево із початкової «платформи» на будь-яку іншу.

2.2. Вирішення задачі

Для зручності позначимо усі «платформи» як k і пронумеруємо їх від 1 до 4, $1 \leq k \leq 4$.

А кожнен V-подібний елмент будемо ідентифікувати за адресою його батьківського вузла, яка в свою чергу буде визначена за принципом, що позначений на рис.4

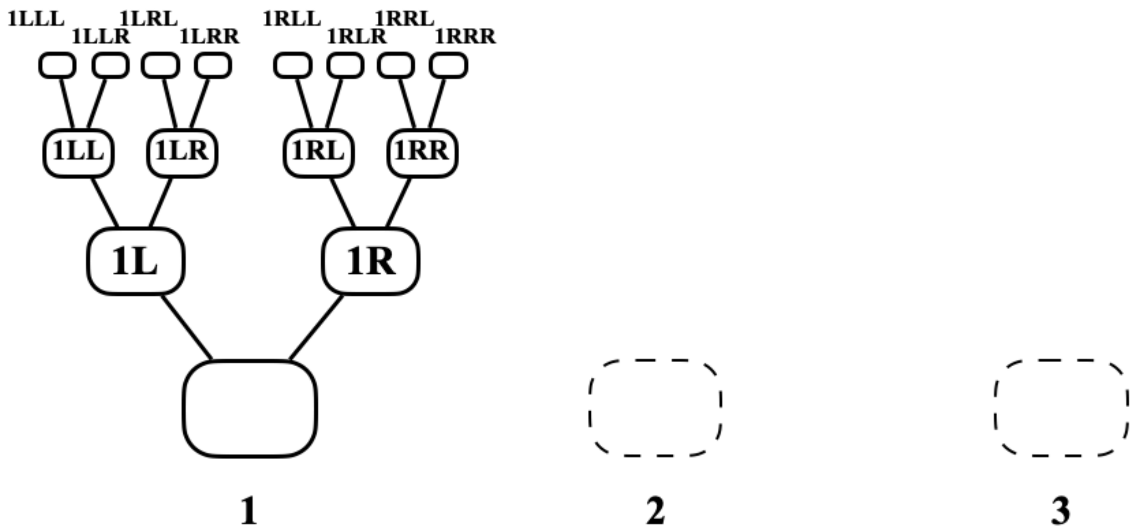


Рис.4 Адреси батьківських вузлів

Так, кожна адреса потребує m ($1 \leq m \leq n - 1$, де n - висота дерева) символів на позначення, залежно від рівня, на якому розташувався сам вузол. Адреса кореневого вузла дерева позначаться номером платформи k, на якій воно знаходиться на даний момент. Усі адреси синівських вузлів є ланцюжами маршрутів до них від кореневого вузла, і складаються із символів із множини Di $\in$ {L, R}, де L відповідає «лівому», а R «правому» розташуванню V-піддерева відносно батьківського вузла елемента.

Кожен хід гри запишемо як *move(x; y)*, де x - адреса V-подібного елемента, що переміщується, а y - його нова адреса, відповідно. Так, до прикладу, переміщення крайнього правого піддерева, розташованого на 3-му рівні на другу «платформу» ми запишемо як (1RR; 2). Ця ж дія схематично зображена на (рис.1)

Розв’язання:

Очевидно, що на перших кроках ми можемо оперувати лише V-елементами верхнього рівня, аж допоки не «розберемо» стек, щоб нам відкрився доступ до елементів нижчих рівнів.

За умови, якби нам була доступна необмежена кількість вільних «платформ», найпростішою стратегією було би розібрати увесь стек, щоб кожен елемент стояв на окремому місці, і почати збирати дерево заново на вільній кінцевій платформі. Але оскільки кількість вільних «платформ» за умовою задачі завжди обмежена (в нашому випадку $\leq$ 3), а верхнерівневих елементів на початку гри $2^{(n - 1)}$, у випадку n = 3 очевидно, що «платформ» не буде вистачати навіть, щоб «зняти» усі елементи найвищого рівня. Звідси ж виникає думка, що якби платформ було занадто мало, задача була би не розв’язною взагалі.

Отже, за перші 3 кроки оптимально буде витратити 2 вільні платформи на 2 елементи найвищого рівня, і ще одну на їхнє спільне батьківське піддерево.

Крок 1. (1LL; 2)

Крок 2.(1LR; 4)

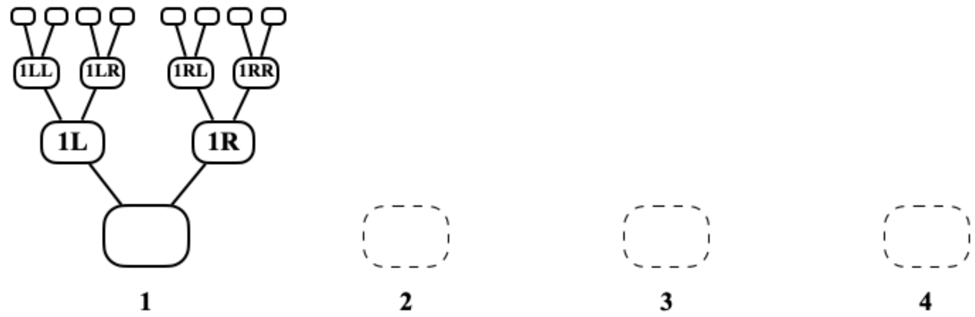
Крок 3.(1L; 3)

Наступними кроками буде поєднати найменші переміщені піддерва на середньому переміщеному. І додатково «зняти» з початкового дерева ті 2 найменші з елементів, що залишились.

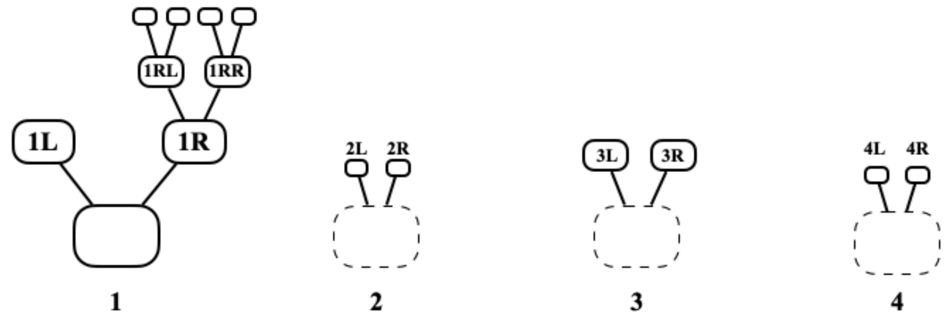
Крок 4. (1RL; 1L)

Крок 5. (1RR; 3L)

Початкова конфігурація задачі:



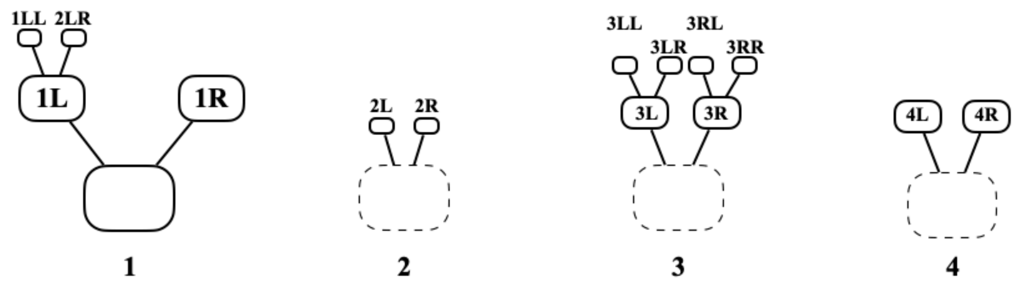
Після перших 3 кроків:



Крок 6.(4; 3R)

Крок 7.(1R; 4)

Після кроків 4-7:

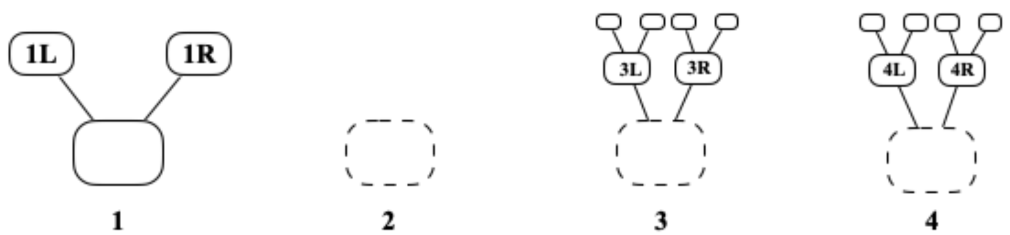


Далі необхідно приєднати на 4L і 4R два найменших елементи.

Крок 8. (2; 4L)

Крок 9. (1L; 4R)

Після кроків 8-9:



Наступними кроками ми перемістимо «основу» початкового графа на її кінцеве місце (2 платформу), та почнемо зворотній процес розбирання піддерев, що поки що переміщені на 3 і 4 платформи і збирання їх на основному корені.

Крок 10. (1; 2)

Крок 11. (3L; 2L)

Крок 12. (3R; 1)

Крок 13. (3; 2R)

Крок 14. (2L; 2RL)

Крок 15. (4L; 2RR)

Крок 16. (4R; 3)

Крок 17. (4; 2L)

Крок 18. (1; 2LL)

Крок 19. (3; 2LR)

Переміщення повного бінарного дерева з 1 в 2 виконано.

Отже, для переміщення повного бінарного дерева висоти 3 на сусідню платформу мінімально знадобилося 19 кроків.

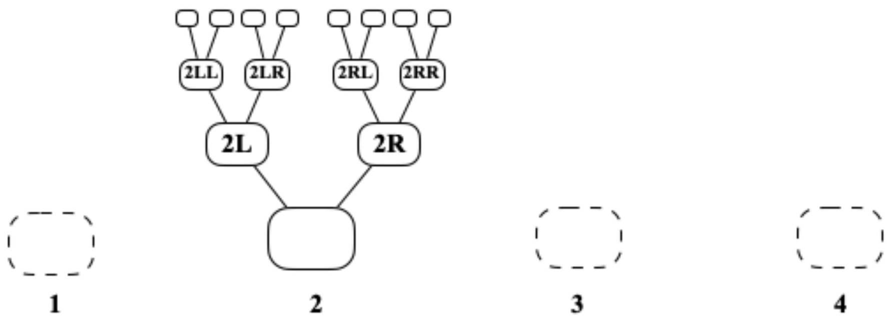


Рис. 8 Гру завершено. Результат після 19 кроку

2.3 Складність обчислення узагальненої задачі

Для загального випадку із деревами висоти n можна записати, що для них існує 2^{i-1} вузлів, однакових між собою на і-тому рівні, $1 \leqslant i \leqslant n$, і таких, що їхній розмір зменшується до кожного наступного рівня. Зауважимо, що S - загальна кількість вузлів у повному бінарному дереві висотою n дорівнює $\sum_{i=0}^{n} 2^i$, а

кількість V-подібних елементів, які потрібно перемістити у кінцеве положення протягом гри = (S - 1))/2 = 2^n - 1

де 1 - кореневий вузол, який ми згідно правил не рухаємо сам по собі, а операція ділення на 2 пояснюється тим, що за один крок переміщається 2 листки одночасно.

У своїй статті автор розглядає лише випадки із k=3 (кількістю вільних платформ) та доводить, що кількість необхідних кроків для розв'язку такої задачі квадратична до загальної кількості заданих вузлів [4]

3.Задача для повних бінарних двокольорових дерев

Ускладнимо задачу із попереднього розділу, додавши вузлам крім їхнього розміру додатковий параметр - колір.

3.1 Формулювання нової задачі

Нехай усі вершини графа будуть розфарбовані [6] у чорний або червоний, при чому так, що вершини одного рівня матимуть однаковий колір. При цьому початковий вузол усього дерева завжди буде чорним, а далі розфарбування рівнів чергуватиметься (Додаток 1. Рис. 1) Тоді до задачі додається умова, що при переміщенні V-подібного піддерева будьякий син не повинен вставати на батька свого ж кольору. «Платформи» вважаються такими, що автоматично приймають колір, протележний до елементів, які на них встають, тобто будь-який елемент може встати на будь-яку порожню «платформу» .

Інші умови задачі з Розділу 2 залишаються незмінними (K = 4, n = 3).

3.2 Перевірка на розв'язність задачі з двокольоровим деревом

Розпочнемо розв'язок за вже відомим оптимальним алгоритмом, враховуючи нові обмеження:

Крок 1. (1LL;2)

Крок 2. (1LR;4)

Крок 3. (1L;3)

Крок 4. (1RR;3R)

Крок 5. (1RL;3L)

На 6 кроці доступними для переміщень є тільки такі V- подібні піддерева: 1R, 2, 3L, 3R і 4. Оптимальним кроком здавалося б переміщення найменшого червоного піддерева 2, але на позицію 1L воно не може потрапити через співпадіння за кольором, а на позиції 3L, 3R та 4 через невиконання умови про розмір батьківських вузлів. Піддерево 4 ізоморфне [5] до піддерева 2, тож ситуація із ним аналогічна. Листки підерев 3L, 3R також червоні і найменші за розміром у початковому графі. Отже, їх можна хіба що повернути на їхнє попереднє місце, що вочевидь не має ніякого сенсу.

Доступ до необхідного переміщення 1R заблокований через брак вільних місць. (Додаток 1. Рис.7)

З чого робимо висновок, що не за будь-якої конфігурації початкових умов задача на повному бінарному двокольоровому дереві є розв'язною.

За конфігурації (K = 4, n = 3) і з розфарбуванням дерева, коли кольори кожного рівня чергуються (за типом червоно-чорних дерев) очевдно, що ми потребуємо додаткову вільну платформу для переміщення наших об'єктів.

3.3 Вдосконалення умов та вирішення задачі

Видозмінимо задачу із Розділу 3.1, додавши додаткову вільну платформу, k= 5. (Рис. 10)

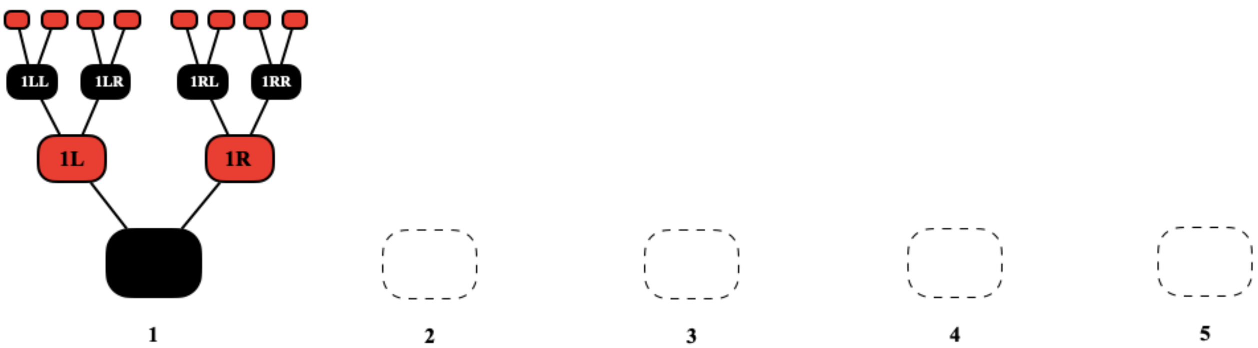


Рис. 10. Задача для двокольорового бінарного дерева при k = 5

Спробуємо вирішити задачу тепер, застосовуючи уже відомий алгоритм. Ілюстрація процесу (Додаток 2).

Кроки 1-5 залишаться такими ж, як і в попередньому розділі. (Додаток 1)

Крок 6. (1R; 5)

Крок 7. (2; 5L)

Крок 8.(1; 2)

Крок 9. (5L; 1)

Крок 10. (5; 2L)

Крок 11. (3L; 2LL)

Крок 12. (3R; 2LR)

Крок 13. (3; 2R)

Крок 14. (4; 2RL)

Крок 15.(1; 2RR)

Дерево повністю переміщено на 2 платформу, отже задачу можна вирішити за 15 кроків.

3.4 Вирішення задачі для однокольорового дерева за новими умовами

Спробуємо перевирішити задачу із Розділу 2.1 за нових умов (тобто введемо додаткову платформу у початкову конфігурацію задачі), k=5. Та порівняємо із розв’язком такої ж двокольорової задачі.

Нова початкова конфігурація зображена на (рис. 11)

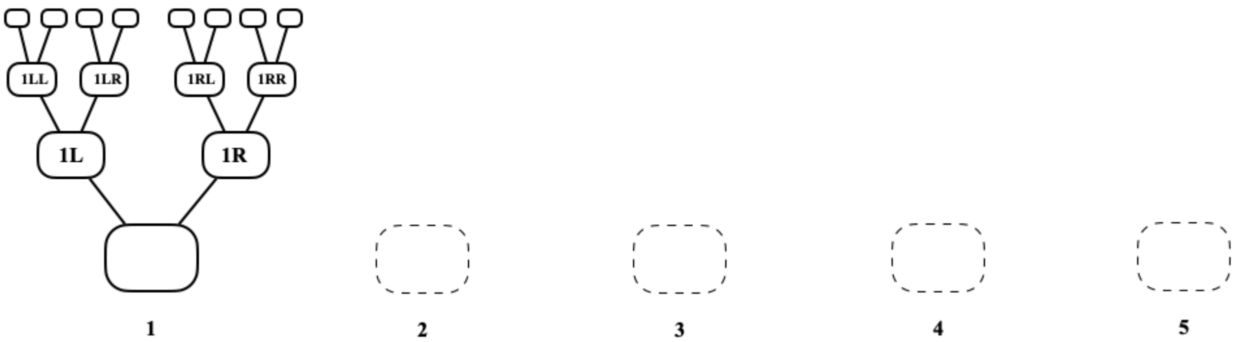


Рис. 11 Задача Енгельфріета із k=5

Крок 1. (1LL;2)

Крок 2. (1LR;4)

Крок 3. (1L;3)

Крок 4. (1RR;3R)

Крок 5. (1RL;3L)

Крок 6. (1R; 5)

Крок 7. (2; 5L)

Крок 8.(1; 2)

Крок 9. (5L; 1)

Крок 10. (5; 2L)

Крок 11. (3L; 2LL)

Крок 12. (3R; 2LR)

Крок 13. (3; 2R)

Крок 14. (4; 2RL)

Крок 15.(1; 2RR)

Дерево повністю переміщено на 2 платформу, задачу також можна вирішити за 15 кроків.

Отже, алгоритм двокольорової задачі повністю спрацював і для однокольорової за умови наявності додаткової платформи. Задача для однокольорового дерева і для двокольорового у цьому частковому випадку ідентичні у своїх розв’язках.

4.1. Додаток 1. Ілюстрація процесу розв’язання задачі про Ханойську вежу для повного бінарного двокольорового дерева висоти 3 для 4 платформ

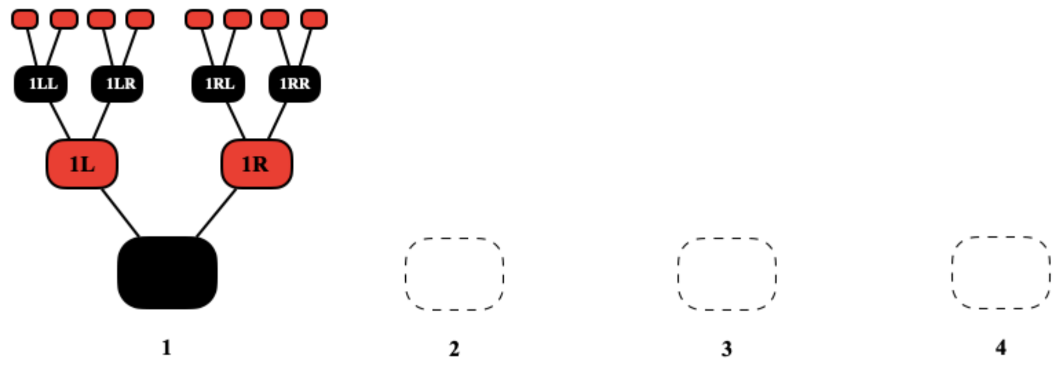


Рис. 1 Початкова конфігурація задачі

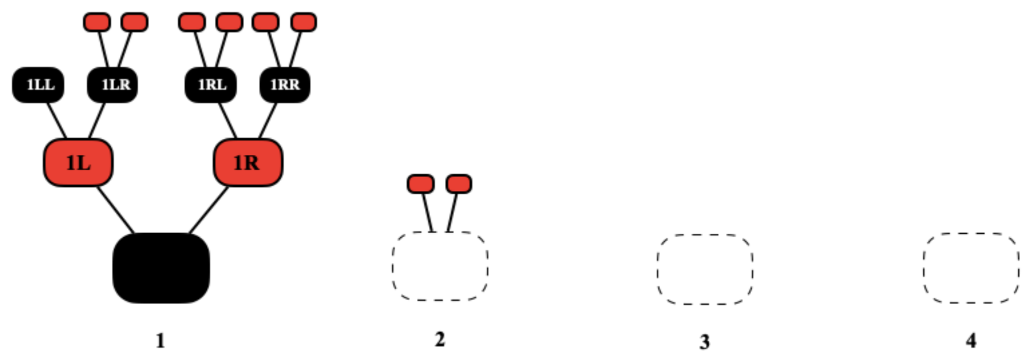


Рис. 2. Крок 1

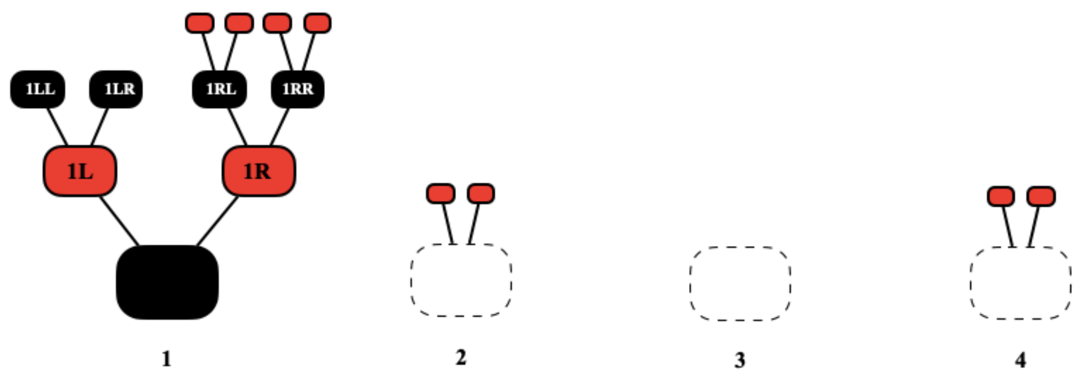


Рис. 3. Крок 2

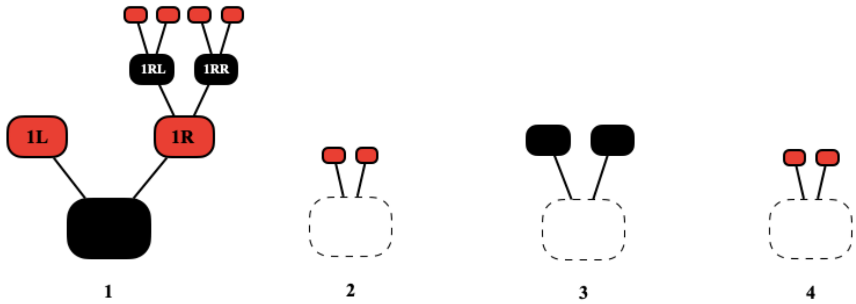


Рис. 4. Крок 3

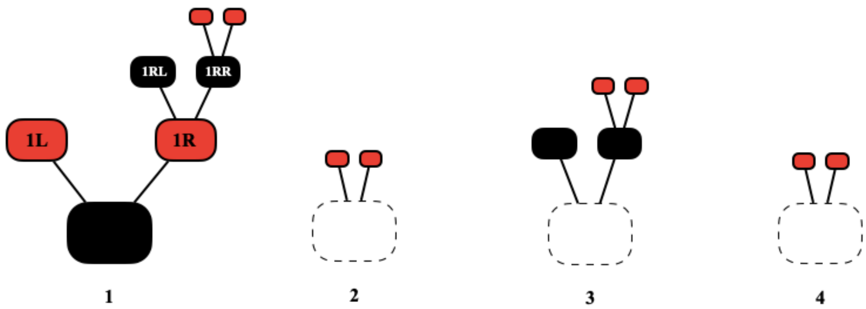


Рис. 5. Крок 4

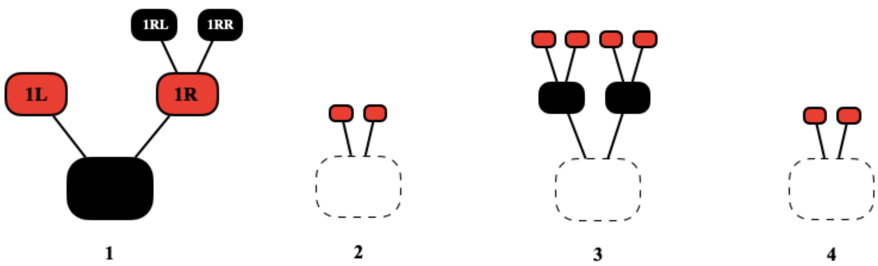


Рис. 6. Крок 5

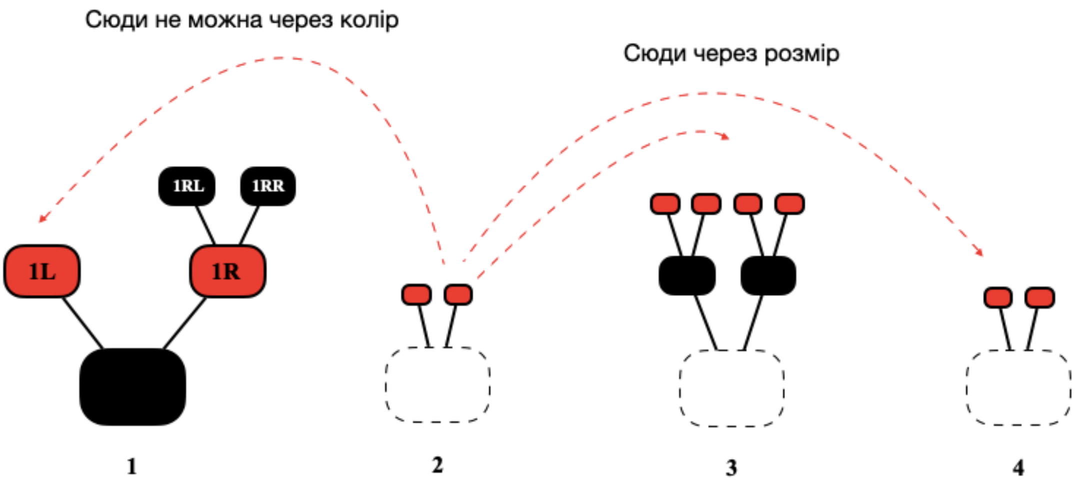


Рис. 7. Задача розв’язків не має

4.2. Додаток 2. Ілюстрація процесу розв'язання задачі про Ханойську вежу для повного бінарного двокольорового дерева висоти 3 для 5 платформ

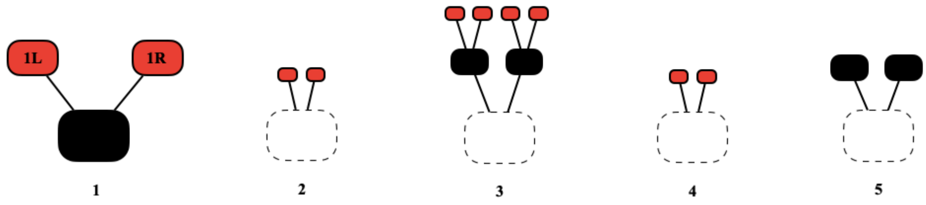


Рис. 1. Крок 6

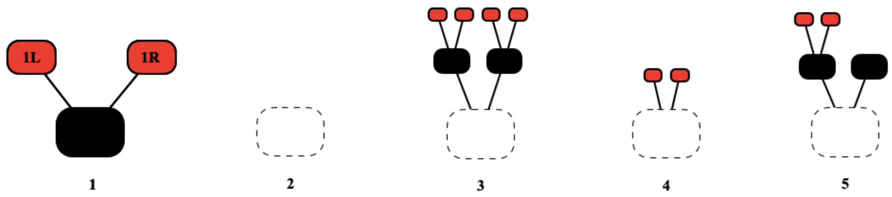


Рис. 2. Крок 7

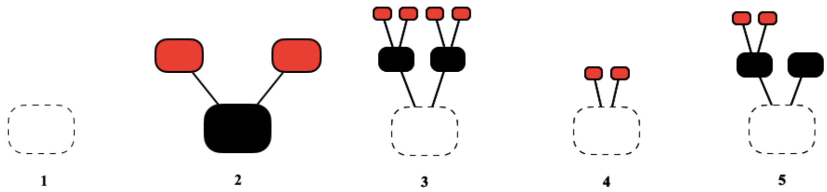


Рис. 3. Крок 8

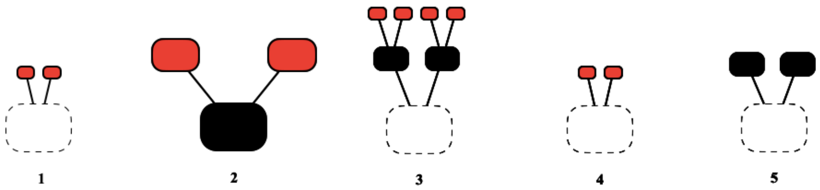


Рис. 4. Крок 9

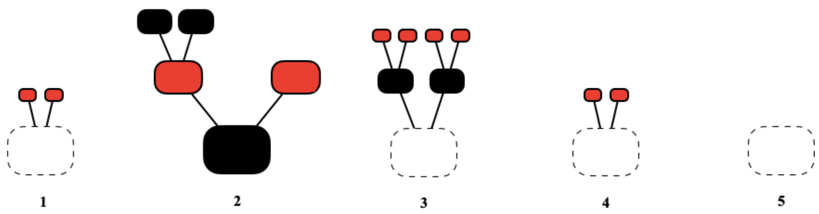


Рис. 5. Крок 10

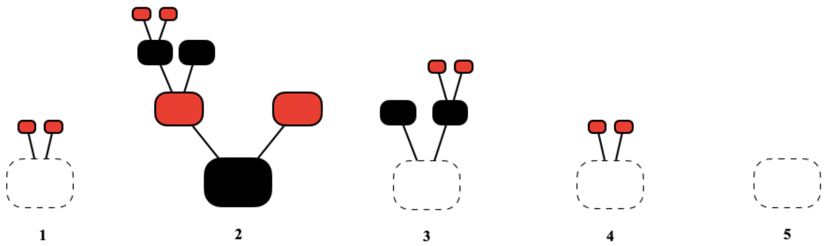


Рис. 6. Крок 11

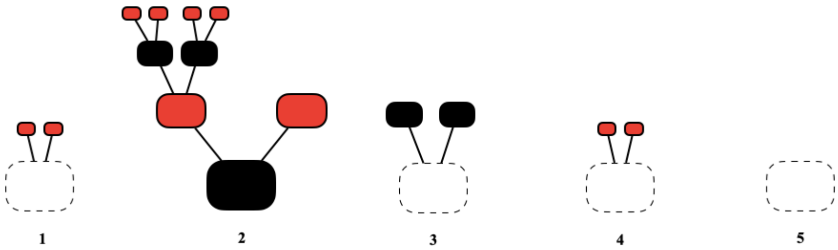


Рис. 7. Крок 12

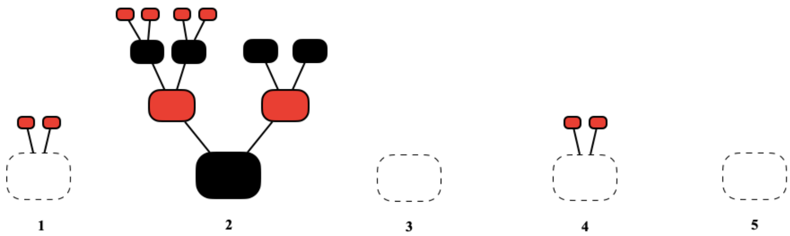


Рис. 8. Крок 13

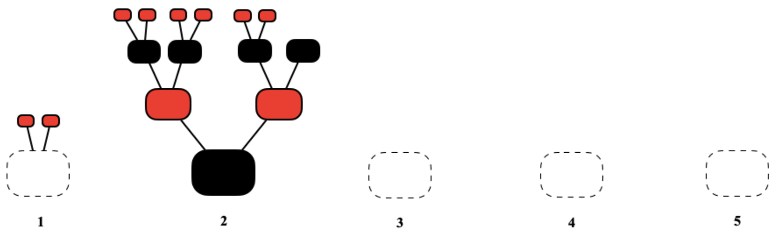


Рис. 9. Крок 14

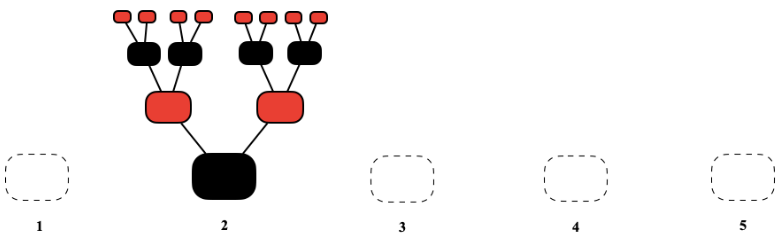


Рис. 10. Крок 15. Кінець розв'язку

4. Висновки

В курсовій роботі розглянуто модифікацію задачі про Ханойську вежу для повних бінарних дерев та оптимальну стратегію її розв’язку для дерева фіксованої величини.

Досліджено частковий випадок задачі для повного бінарного двокольорового дерева, розфарбованого за схемою червоно-чорних дерев. В результаті чого виявлено, що від конфігурації початкових умов задачі залежить її розв’язність, і емпіричним шляхом підібрана така конфігурація гри, що задача є розв’язною.

Також виявлено, що у випадку використання 5 платформ для переміщень елементів, задача для двокольорового дерева висоти 3 розв'язується ідентично до задчі з однокольоровим бінарним деревом.

Напрямок цього дослідження можна продовжувати розвивати і знайти, наприклад, узагальнений розв’язок задачі з бінарними двокольоровими деревами та порівняти його з узагальненим розв’язком для однокольорових дерев. Також можна порахувати кількість необхідних кроків для дерев більшої висоти, із використанням іншої кількості кольорів і схем розфарбування, тощо.

6. Список використаної літератури

1. Petković, Miodrag. Famous Puzzles of Great Mathematicians, 197 p. (2009)

2. Санжаровська А. О.: Узагальнення задачі про Ханойську вежу (2017)

3. J¨orgensen M.E.: Solutions to the generalized Towers of Hanoi problem (2012)

4. Engelfriet J.: The Trees of Hanoi (1981)

5. Гавриленко О.: Дерева. Дискретна математика 265 с., 157-163 (2002)

6. Боднарчук Ю.В., Олійник Б.В.: Основи дискретної математики 138с, 125-130 (2007)