



ФОРМАЛІЗАЦІЯ ПОСТАНОВОК ЗАДАЧ ПРО УКЛАДАННЯ ТУРИСТИЧНОГО РАНЦЯ ТА АЛГОРИТМИ ЇХ РОЗВ'ЯЗАННЯ

Наведено формалізовані постановки задач про укладання туристичного ранця, запропоновано ефективні алгоритми їх розв'язання, що загалом дало змогу отримати адекватні результати розрахунку, провести змістовний їх аналіз та вибрати вдалі постановки задач для їх подальшого застосування. З'ясовано, що процедура укладання туристичного наплічника зазвичай є проблемою як для початківців, так і бувалих мандрівників. Водночас, досвідчені туристи в таких ситуаціях володіють деякими загальними правилами, які дають їм змогу вкладати найбільш потрібні речі не тільки встановленої місткості та мінімальної ваги, але й дотримуватись деякого порядку розміщення цих речей в наплічнику і надати йому традиційну форму, що забезпечує зручність тривалого його перенесення. Виявлено, що класична постановка задачі про ранець належить до задач цілочисельного програмування, вона допускає значну кількість різних узагальнень залежно від обмежень, накладених на ранець, на предмети або на їх вибір, а також на умову отримання оптимального розв'язку задачі – булевого чи кількісного. Проаналізовано можливі варіанти її постановок, з'ясовано основні причини широкого застосування в різних областях знань. Встановлено, що задача про ранець належить до класу NP-повних задач комбінаторної оптимізації, тому для неї немає поліноміального алгоритму, здатного її розв'язати за розумний проміжок часу. Визначено особливості застосування точних методів розв'язання задачі про ранець, проаналізовано метод повного перебору можливих варіантів, метод гілок і меж, жадібний алгоритм і методи динамічного програмування. Дано рекомендації щодо вибору серед них найпридатнішого для розв'язання запропонованих у роботі постановок задач. Наведено приклади деяких практичних постановок задачі про ранець, здійснено їхню формалізацію, алгоритмізацію та програмну реалізацію, запропоновано адекватний метод розв'язання, а також проведено змістовний аналіз отриманих результатів розрахунку, на підставі яких вибрано вдалі постановки задач для їх подальшого застосування.

Ключові слова: задача про ранець; задача цілочисельного програмування; задачі комбінаторної оптимізації; обмеження та умови задачі; умова оптимального розв'язку задачі.

Вступ. Туристичний похід – спланована подорож організованої групи осіб, або одного мандрівника з використанням активних форм пересування за визначеним маршрутом, під час проходження якого можливе подолання природних перешкод – перевалів, порогів, печер різних категорій та ступенів складності. Перед тим, як піти в похід, потрібно ретельно підготуватися до нього. Досвідчені туристи можуть нехтувати деякими етапами підготовки до походу, але для початківців це не припустимо. Оскільки основне завдання туристичного походу – сприяння активному відпочинку, зміцнення здоров'я, набуття прикладних і спеціальних навиків, то організувати його потрібно так, щоб похід був зручним для ходьби і приносив задоволення кожному з його учасників.

Основною проблемою при плануванні походу для початківці є укладання туристичного наплічника, адже він має обмежену місткість та вагу, яку може переносити людина. Кожен досвідчений турист знає, що найголовніше при укладанні наплічника – це не брати нічого зайвого. Тому під час планування походу важливо виз-

начити фізичний стан кожного з його учасників, позаяк їм доведеться постійно долати різні фізичні навантаження – від перенесення наплічника до кардіо-навантажень під час підйому на вершину.

Свою назву задача про укладання туристичного наплічника (задача про ранець, англ. *Knapsack Problem*) отримала від кінцевої мети: укласти якомога більше цінних предметів у ранець за умови, що його місткість обмежена. У загальному вигляді задачу про ранець можна сформулювати так: із заданої множини предметів з властивостями "вартість" і "вага" потрібно відібрати підмножину з максимальною вартістю, дотримуючись при цьому обмеження ранця на загальну вагу (рис. 1).

Достеменно невідомо, хто першим навів математичне формулювання задачі про ранець. Одну з перших згадок про неї можна знайти в статті Джорджа Метьюса Балларда (Mathews, 1897; Kellerer, Pferschy & Pisinger, 2004, p. 3), датовану 1897 роком. Інтенсивне вивчення цієї проблеми почалося після публікації Д. Б. Данцигом у 1957 році книги "Дискретно-змінні екстремальні проблеми" (англ. *Discrete Variable Extremum Problem*)

Інформація про авторів:

Грицюк Юрій Іванович, д-р техн. наук, професор, кафедра програмного забезпечення.

Email: yurii.i.hrytsiuk@lpnu.ua; <https://orcid.org/0000-0001-8183-3466>

Немова Олена Андріївна, студентка, кафедра програмного забезпечення. **Email:** olena.nemova@gmail.com

Цитування за ДСТУ: Грицюк Ю. І., Немова О. А. Формалізація постановок задач про укладання туристичного ранця та алгоритми їх розв'язання. Науковий вісник НЛТУ України. 2019, т. 29, № 4. С. 93–102.

Citation APA: Hrytsiuk, Yu. I., & Nemova, O. A. (2019). Formalization of task statements for a knapsack problem and algorithms to solve them. *Scientific Bulletin of UNFU*, 29(4), 93–102. <https://doi.org/10.15421/40290420>

(Dantzig, 1957; Kellerer, Pferschy & Pisinger, 2004, p. 9), особливо в 1970-90-і роки як видатними теоретиками, так і знаними практиками (Martelo & Toth, 1990, p. 2). Задача про ранець є однією з 21 NP-повних задач Річарда Карпа (англ. *Richard Karp*), викладених у його статті під назвою "Reducibility Among Combinatorial Problems" в 1972 році (Karp, 1972).

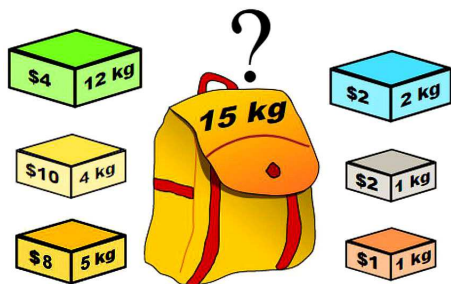


Рис. 1. Приклад задачі про ранець (Knapsack problem, 2019)

Значний інтерес до цієї задачі був викликаний достатньо простим її формулюванням, великою кількістю її різновидів, властивостей і, водночас, складністю алгоритмів їх розв'язання. Відома на сьогодні більшість алгоритмів на практиці здатні розв'язувати різні постановки задач досить великих розмірів (Dantzig, 1957). Однак, унікальне формулювання задачі, а також той факт, що вона присутня як підзадача у більших, дещо загальніших проблемах, робить її важливою для проведення нових наукових досліджень.

Отже, для багатьох туристів-початківців процедура укладання наплічника предметами різної величини та ваги не завжди завершується успішно, позаяк пов'язана з відсутністю у них як деяких елементарних знань, так і з невиконанням певних правил порядку і місця їх розміщення, або з перевищенням їх загальної ваги. Тому *актуальним завданням* наразі є формалізація практичних постановок задач про укладання ранця та розроблення ефективних алгоритмів їх розв'язання, а також проведення змістовного аналізу отриманих розв'язків і вибору серед них найбільш перспективних.

Об'єкт дослідження – розв'язання задачі про ранець.

Предмет дослідження – методи та засоби формалізації деяких постановок задач про ранець, які дадуть змогу здійснити їх математичне формулювання, вибрати адекватний метод розв'язання та провести аналіз отриманих результатів.

Метою дослідження є розроблення ефективної методики розв'язання практичних постановок задачі про ранець, яка дасть змогу отримати адекватні результати розрахунку, провести їх змістовний аналіз та вибрати вдалі постановки для подальшого застосування.

Для реалізації зазначеної мети потрібно виконати такі основні завдання:

- 1) в'яснити, чому укладання туристичного наплічника зазвичай є проблемою для початківців, а досвідчені туристи володіють деякими загальними правилами, які дають їм змогу вкладати потрібні речі не тільки встановленої місткості та мінімальної ваги, але й дотримуватись самої форми наплічника для зручності його перенесення;
- 2) з'ясувати особливості класичної постановки задачі про ранець, проаналізувати можливі її варіанти, в'яснити причини її широкого застосування в різних областях знань;

- 3) визначити особливості застосування точних методів розв'язання задачі про ранець, проаналізувати метод повного перебору можливих варіантів, метод гілок і меж, жадібний алгоритм і методи динамічного програмування та вибрати серед них найпридатніший для розв'язання запропонованих постановок задач;
- 4) навести приклади деяких практичних постановок задач про ранець, запропонувати ефективні методики їх розв'язання, провести змістовний аналіз отриманих результатів розрахунку, на підставі яких вибрати вдалі постановки задач для подальшого їх застосування;
- 5) зробити відповідні висновки та надати рекомендації щодо використання розробленого ПЗ для розв'язання різних постановок задач про укладання ранця.

1. Проблема укладання туристичного наплічника.

Правильне укладання наплічника має велике значення, незалежно від того, чи йде турист в багатоденний похід або на одноденну прогулянку. Головна вимога при укладанні наплічника: за умови найменших зовнішніх габаритів він має володіти найбільшою місткістю і бути зручним для перенесення. Для багатоденного походу підійдуть великі наплічники місткістю 50-70 літрів, які виготовлені зі щільної та міцної тканини, мають ущільнене дно і до яких надійно прикріплені широкі наплічні лямки.

При укладанні наплічника досвідчені туристи дотримуються такого загального правила: важкі речі кладуть вниз або всередину і ближче до спини; м'які – до спини; об'ємні та крихкі – вгору; речі першої потреби – в кишені наплічника. Варто укласти речі в наплічник так, щоб при русі вони не гриміли між собою. Також важливо запам'ятати місце розташування різних речей в наплічнику, щоб за потреби деяку річ можна швидко дістати, а не порпатися в наплічнику (рис. 2).



Рис. 2. Варіант укладання речей в наплічник

Досвідчені туристи спочатку вкладають в наплічник згорнутий килимок, щоб надати йому форму. Після цього до спини укладають спальник або ковдру, загорнуті у вологостійкий пакет (зазвичай пакет для сміття), інші великі та м'які предмети, потім – важкі консерви, цукор, крупи. Зверху наплічника і в його кишені кладуть все те, що може першочергово знадобитися в дорозі. Особистий змінний одяг та документи також потрібно загортати у водонепроникні пакети.

Для зручності укладання наплічника та швидкого знаходження потрібних речей зазвичай роблять так: у один великий поліетиленовий пакет кладуть спальний мішок та всі спальні предмети, в другий – весь одяг, в третій – продукти харчування, а дрібні речі кладуть окремо. Доцільно дотримуватися певної системи при ук-

ладанні наплічника і при його розбиранні: кожна річ як у наплічнику, так і всередині намету повинна мати своє певне місце.

Коли укладання наплічника завершено, треба перевірити його форму та довжину лямок. Не варто сильно затягувати лямки: вдало запакований і збалансований наплічник буде добре сам прилягати до спини, а з затягнутими лямками йти дуже важко. І ще одне: ідеально складений наплічник, якщо його поставити вертикально, не повинен падати вбік або назад, а тільки вперед.

Після укладання всіх речей в наплічник, туристу варто напнути його на себе, щоб перевірити, чи зручно його нести, чи не впирається щось тверде в спину. Правильно укладений наплічник повинен щільно прилягати до спини увігнутою стороною. Не варто прикріплювати крупні предмети ззовні наплічника або знизу його. Позаяк уникнути цього неможливо, доцільно такі предмети, як намет, спальник чи каремат заздалегідь загорнути у водонепроникний пакет і щільно прив'язати до наплічника. Під клапан наплічника зазвичай кладуть штормівку, легку куртку, накидку чи дощовик. Окрім змінного одягу і предметів гігієни, в піший похід потрібно обов'язково брати взуття (краще черевики) на міцній підошві. Для довгого походу обов'язково потрібно взяти запасну білизну і шарпетки. Навіть якщо на вулиці літо і вдень чимала жара, потрібно взяти теплу вологостійку куртку і шапочку, позаяк у горах ночі холодні, а під ранок часто бувають й заморозки.

Варто пам'ятати, що туристичне спорядження – предмети, які використовують під час походу, тому вони мають бути надійними, легкими, міцними і зручними. Перелік спорядження зазвичай залежить від сезону, погоди, маршруту, його складності та тривалості. Від того, яке спорядження буде у туриста, залежить його здоров'я та успішність перебігу самого походу. А вже в похід варто брати не те, що потрібно, а те, без чого неможливо обійтись. Зайві речі тільки збільшують вагу наплічника. Наприклад, якщо сім членів групи візьмуть по тюбику зубної пасти (100 г), по куску звичайного (150 г) і господарського мила для прання речей (200 г), туалетний папір, шампунь і т.д., то зайвими виявляться більше 3 кг непотрібного вантажу. Для уникнення подібного дублювання речей потрібно домовитися, хто візьме на себе відповідальність за туалетне приладдя, яке взяти мило і зубну пасту. Такі загальні речі можна розподілити між членами групи. Зазвичай, пакет з ними під час стоянок вішають на видному місці для загального користування.

2. Класична постановка задачі про ранець. Проблема укладання туристичного наплічника змістовно означає відбір предметів з найбільшою їх загальною цінністю та обмеженою його місткістю. Така проблема також часто виникає при виборі оптимального процесу управління в різних економіко-фінансових ділянках (розподіл бюджету відділу за проектами і т. п.). Задачі такого типу належить до задач цілочисельного програмування (Sigal & Ivanova, 2007), найпростішою серед яких є задача про ранець.

Назва "задача про ранець" пов'язана з інтерпретацією задачі відбору найкращого складу предметів, що задовольняють проблему туриста щодо оптимальної кількості речей. Турист, йдучи в похід, має відібрати із наявного переліку найпотрібніші предмети, кожен з яких має два параметри – вагу і цінність. Ранець туриста

має певну місткість. Завдання полягає в тому, щоб укласти в ранець різні предмети з максимальною їх загальною цінністю, дотримуючись при цьому обмеження на загальну вагу.

Математично задача має таке формулювання (Martelo & Toth, 1990, p. 1): задано n предметів. Для i -го предмету встановлено його вагу $w_i > 0$ і цінність $v_i > 0, i = \overline{1, n}$. Обмеження загальної ваги предметів у ранці задається його місткістю W . Необхідно максимізувати цінність предметів у ранці

$$F = \sum_{i=1}^n v_i x_i \rightarrow \max, \quad (1)$$

з дотриманням обмеження на місткість ранця $\sum_{i=1}^n w_i x_i \leq W$, а також за умови отримання булевого розв'язку задачі $x_i \in \{0, 1\}, i = \overline{1, n}$.

Варіанти задачі про ранець. Постановка задачі про ранець допускає значну кількість різних узагальнень залежно від обмежень, накладених на ранець, на предмети або на їх вибір, а також на умову отримання оптимального розв'язку задачі – булевого чи кількісного. Найбільш популярними є такі різновиди задачі:

- 1) *Ранець 0-1* (англ. *0-1 Knapsack Problem*) (Martelo & Toth, 1990, p. 2) – укладання в ранець не більше одного предмету різного виду.
- 2) *Обмежений ранець* (англ. *Bounded Knapsack Problem*) (Kellerer, Pferschy & Pisinger, 2004, p. 127) – укладання в ранець не більше певної кількості предметів різного виду.
- 3) *Необмежений ранець* (англ. *Unbounded Knapsack Problem*) (Kellerer, Pferschy & Pisinger, 2004, p. 127) – укладання в ранець довільної кількості предметів різного виду.
- 4) *Ранець з мультивибором* (англ. *Multiple-Choice Knapsack Problem*) (Kellerer, Pferschy & Pisinger, 2004, p. 147), де предмети різного виду розділені на групи, із кожної з яких потрібно вибрати тільки один предмет певного виду і вкласти в ранець.
- 5) *Множинний ранець* (англ. *Multiple Knapsack Problem*) (Martelo & Toth, 1990, p. 157) коли в декілька ранців, кожен з яких має свою максимальну місткість, потрібно вкласти предмети певної кількості різного виду.
- 6) *Багатовимірний ранець* (англ. *Multy-Dimensional Knapsack Problem*), коли замість ваги предметів задано декілька різних ресурсів (наприклад, вага, обсяг і тривалість укладання). Кожен предмет витрачає задану кількість кожного ресурсу. Треба вибрати підмножину предметів так, щоб загальні витрати кожного ресурсу не перевищували його наявного обсягу, і при цьому загальна цінність предметів була максимальною (Kellerer, Pferschy & Pisinger, 2004, p. 147).
- 7) *Квадратична задача про ранець* (англ. *Quadratic Knapsack Problem*), коли загальна цінність предметів задано невід'ємно певною квадратичною формулою (Gallo, Hammer & Simeone, 2009).

Наочна інтерпретація задачі про ранець привела до того, що вона знайшла широке застосування в різних областях знань: в математиці, інформатиці та на стику цих наук – в криптографії. На підставі задачі про ранець був створений перший алгоритм асиметричного шифрування. Ще в 1976 році ідея криптографії з відкритими ключами була вперше представлена Уїтфілдом Діффі і Мартіном Хеллманом на Національній комп'ютерній конференції (англ. *National Computer Confe-*

rence) (Shnaier, 2002, p. 19.2; Gabidulin, Kshevetckii & Kolybelnikov, 2011). У галузі криптоаналізу в 1977 році було опубліковано метод "зустріч посередині" та запропоновано відповідну атаку (англ. Meet-in-the-Middle Attack), яку названо класом атак на криптографічні алгоритми, що асимптотично зменшують тривалість повного перебору за рахунок принципу "розділяй і володарюй", однак значно збільшують об'єм необхідної пам'яті (Martelo & Toth, 1990, p. 1).

В одній з робіт з обчислювальної лінгвістики (Riedhammer et al., 2008, pp. 24–36) було запропоновано формулювання задачі *автоматичного реферування текстів*, окремий випадок якої відповідає постановці задачі про ранець. Також ця задача може слугувати моделлю для вирішення великої кількості промислових завдань (Martelo & Toth, 1990, p. 1; Burkov, Gorgidze, & Lovetckii, 1974), а саме:

- *розміщення різних предметів* на складі заданої площі, де потрібно максимально заповнити площу складу предметами максимальної цінності;
- *розкрій тканини*, де з наявного куска матеріалу потрібно отримати максимальну кількість викрійок певної форми, дотримуючись обмеження на текстуру тканини;
- *розрахунок оптимальних капіталовкладень*, де під час планування подальшої діяльності виробничої системи, яка складається з декількох підприємств, необхідно визначити, яку суму коштів виділити кожному з них, щоб отримати максимальний загальний прибуток від їхнього використання.

3. Точні методи розв'язання задачі про ранець.

Ключовою проблемою сучасної теорії управління є оптимізація виробничих процесів. Вирішення цієї проблеми вимагає розроблення та практичного застосування точних методів оптимізації, що ґрунтуються на використанні сучасної комп'ютерної техніки. Серед прикладних задач оптимізації важливе місце займають задачі дискретного програмування. При цьому розрізняють задачі комбінаторного типу, допустима множина яких має скінченну кількість точок, задачі цілочисельного програмування, де змінні набувають цілочисельні значення, та задачі частково дискретного програмування, в яких тільки частина змінних набуває дискретні значення.

Як було зазначено вище, задача про ранець належить до класу NP-повних задач комбінаторної оптимізації, і для неї немає поліноміального алгоритму, здатного її розв'язати за розумний проміжок часу. Тому під час розв'язання цієї задачі досліднику необхідно вибрати між точними алгоритмами, які не придатні для вирішення проблеми "великих" ранців, і наближеними, які працюють швидко, але не гарантують отримання оптимального розв'язку задачі.

Спробуємо проаналізувати точні методи розв'язання задачі про ранець і дати свої рекомендації щодо вибору кращого з них за деякими критеріями – "величина ранця" (обсяг даних), точність і тривалість розв'язання.

Повний перебір можливих варіантів. Як і для багатьох інших дискретних задач, задачу про ранець можна розв'язати, перебравши повністю всі можливі варіанти її розв'язку. Припустимо, що є n предметів, які можна укласти в ранець. Потрібно укласти предмети в ранець так, щоб їхня вартість була максимальною, водночас його місткість не перевищувала заданої величини W .

Для кожного предмету існує два варіанти відбору: предмет або кладуть в ранець, або ні. Тоді перебір всіх можливих варіантів відбору має тимчасову складність

$O(2^n)$, що дає змогу його використовувати тільки для невеликої кількості предметів (Okulov, 2007, pp. 92–93]. Зі зростанням їх кількості задача про ранець стає нерозв'язною цим методом за прийнятний проміжок часу.

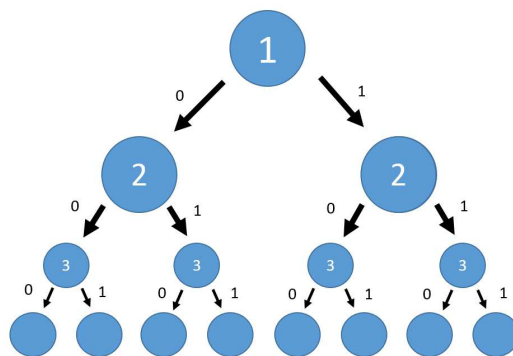


Рис. 3. Дерево повного перебору, що відповідає пошуку розв'язку для трьох предметів (Knapsack problem, 2019)

На рис. 3 показано дерево повного перебору для трьох предметів. Кожен лист відповідає деякій підмножині предметів. Після складання дерева необхідно знайти лист з максимальною цінністю предметів серед тих, вага яких не перевищує задану місткість W (Okulov, 2007, pp. 101–105].

Метод гілок і меж є варіацією методу повного перебору з тією різницею, що свідомо вилучають неоптимальні гілки цього дерева. Як і метод повного перебору, він дає змогу знайти оптимальний розв'язок і тому належить до точних алгоритмів.

Жадібний алгоритм, запропонований Пітером Колесаром (англ. Peter Kolesar) в 1967 році, пропонує спочатку впорядкувати предмети за їх питомою вартістю (стосовно цінності до ваги) і будувати дерево повного перебору. Його поліпшення полягає в тому, що в процесі побудови дерева для кожного вузла потрібно оцінювати верхню межу цінності розв'язку, після чого продовжуємо будувати дерево тільки для вузла з максимальною оцінкою (Martello & Toth). Коли максимальна верхня межа виявляється в листі дерева, алгоритм завершує свою роботу.

Для розв'язання задачі про ранець жадібним алгоритмом, необхідно впорядкувати предмети за їхньою питомою цінністю (тобто за відношенням цінності предмета до його ваги), і помістити в ранець предмети з найбільшою питомою цінністю (Martello & Toth, 1990). Тривалість роботи такого алгоритму складається з тривалості сортування та тривалості укладання. Складність сортування предметів становить $O(\log(N))$. Далі відбувається обчислення того, скільки предметів поміститься в ранець за загальний час $O(N)$.

Здатність методу гілок і меж та жадібного алгоритму зменшувати кількість варіантів повного перебору сильно залежить від вхідних даних. Його доцільно застосовувати тільки в тому випадку, коли питомі цінності предметів значно відрізняються між собою (Diubin & Korbut, 1999).

Методи динамічного програмування зазвичай використовують для розв'язання задачі про необмежений ранець. При додатковому обмеженні на вагу предметів таку задачу можна розв'язати за псевдо-поліноміальний час.

Нехай вага кожного предмета $\tilde{W} = \{w_i, i = \overline{1, n}\}$ є цілим невід'ємним числом. Тоді для розв'язання задачі необхідно обчислити оптимальні розв'язки для всіх

$w \in \mathbb{Z} : 0 \leq w \leq W$, де W – задана місткість ранця. Визначимо $m[w]$ як максимальну цінність предметів, які можна помістити в рюкзак заданої місткості W .

Для $m[w]$ можна записати такі рекурентні формули (Romanovskii, 1977):

$$m[0] = 0; m[w] = \max_{w_i \leq w} \{v_i + m[w - w_i]\},$$

де v_i , w_i – цінність і вага i -го предмета відповідно, а максимум з порожньої множини варто вважати такими, що дорівнюють нулю.

Фактично останнє рівняння є функціональним рівнянням Р. Беллмана, яке ще називають функціональним рівнянням динамічного програмування. Для його розв'язання достатньо обчислити всі значення $m[w]$, починаючи з 0 і до W (Romanovskii, 1977). Якщо додатково зберігати на кожному кроці набір предметів, який реалізує максимальну їх цінність, то алгоритм видасть їхній оптимальний набір.

Позаяк на кожному кроці необхідно знайти максимум з n предметів, то алгоритм має обчислювальну складність $O(nW)$. Оскільки значення W може залежати експоненціально від розміру вхідних даних, то алгоритм є псевдо-поліноміальним, то його ефективність визначають цим значенням W . Алгоритм дає відмінні результати при $W \leq 1000$, але не дуже хороші для $W \geq 10\,000\,000$ (Skiena, 2011).

Отже, для отримання адекватного розв'язку наведених вище постановок задач можна використовувати будь-який метод, однак найбільшу перевагу варто надати жадібному алгоритму, позаяк його програмна реалізація є найлегшою.

4. Приклади розв'язання деяких постановок задач про ранець. Не претендуючи на значні здобутки у вирішенні проблеми розв'язання задачі про ранець, спробуємо внести і свою лепту у формулювання деяких оригінальних постановок задач і запропонуємо ефективні методики їх розв'язання, особливо тих, які стосуються практичного їх застосування. Тому, наведемо приклади деяких практичних постановок задач про ранець, здійснимо їхню формалізацію та алгоритмізацію, виберемо адекватний метод розв'язання, а також проведемо змістовний аналіз отриманих результатів.

Приклад 1: задача про монети. Задано набір монет, для кожної з яких відомі: номінал (1, 2, 5, 10, 25, 50

коп, 1 грн звичайна (100 коп) і 1 грн Вол. Великого (100 коп); вага (1,5, 1,8, 4,3, 1,7, 2,9, 4,2, 7,1 і 6,8 г); кількість (12, 23, 18, 31, 22, 27, 8 і 13 шт). Також є замовлення на відбір певної суми монет – 4153 коп. Завдання полягає в тому, щоб відібрати зазначену суму монет, дотримуючись при цьому обмеження на наявну їх кількість.

Математична постановка задачі. Введемо такі позначення:

- n – кількість номіналів монет;
- $\tilde{W} = \{w_j, j = \overline{1, n}\}$ – номінал j -ої монети, коп;
- $\tilde{V} = \{v_j, j = \overline{1, n}\}$ – вага монети j -го номіналу, г;
- $\tilde{K} = \{k_j, j = \overline{1, n}\}$ – наявна кількість монет j -го номіналу, шт;
- $\tilde{X} = \{x_j, j = \overline{1, n}\}$ – невідома кількість монет j -го номіналу, яку потрібно встановити, шт;
- S – замовлена сума монет, кг.

Тоді задача має таке математичне формулювання: задано n монет певного номіналу (w_j), ваги (v_j) та наявної кількості (k_j). Необхідно відібрати зазначену суму монет певного номіналу

$$\sum_{j=1}^n w_j x_j = S, \quad (2)$$

з дотриманням обмеження на наявну кількість монет j -го номіналу $x_j \leq k_j, j = \overline{1, n}$, а також за умови отримання натурального розв'язку задачі $x_j \in \mathbb{N}_0, j = \overline{1, n}$.

Наведена постановка задачі належить до варіанту задачі про обмежений ранець (англ. *Bounded Knapsack Problem*), де в ранець вказаної місткості потрібно укласти певну кількість предметів різного виду.

Методика розв'язання задачі. Складемо табл. 1.1, в яку внесемо значення всіх вхідних даних і розрахуємо ще деякі додаткові параметри – сумарні вартість і вагу.

Нехай для відбору зазначеної суми спочатку візьмемо по одній монеті кожного номіналу та розрахуємо табл. 1.2. З цієї таблиці видно, що внаслідок виконання таких дій отримаємо суму монет у 293 коп (а потрібно 4153 коп), вага яких становить 30,3 г.

Для отримання зазначеної суми монет відповідного номіналу використаємо певну процедуру оптимізації, дотримуючись при цьому обмеження на наявну їх кількість, внаслідок чого матимемо таку нову табл. 1.3.

Табл. 1.1. Вхідні дані задачі про монети

Характеристики монет	Звичайні копійки						Гривні		Сума наявна	Набрати, коп
	1	2	3	4	5	6	7*	8**		
Номінал, коп	1	2	5	10	25	50	100	100		
Вага, г	1,5	1,8	4,3	1,7	2,9	4,2	7,1	6,8		
Кількість, шт	12	23	18	31	22	27	8	13		
Сума, коп	12	46	90	310	550	1350	800	1300	4458	4153
Сума, г	18	41,4	77,4	52,7	63,8	113,4	56,8	88,4	511,9	

Примітка: * – звичайна гривня; ** – гривня Володимира Великого.

Табл. 1.2. Розрахункові дані задачі про монети перед процедурою оптимізації

Номінал, коп	1	2	5	10	25	50	100*	100**	Сума		
Кількість, шт	1	1	1	1	1	1	1	1			
Сума, коп	1	2	5	10	25	50	100	100	293	коп	-3860
Сума, г	1,5	1,8	4,3	1,7	2,9	4,2	7,1	6,8	30,3	г	
Залишок, коп	11	44	85	300	525	1300	700	1200	4165		

Табл. 1.3. Результати розв'язання задачі про монети після процедури оптимізації

Номінал, коп	1	2	5	10	25	50	100*	100**	Сума		
Кількість, шт	2	3	7	11	22	27	8	13			
Сума, коп	2	6	35	110	550	1350	800	1300	4153	коп	0
Сума, г	3,0	5,4	30,1	18,7	63,8	113,4	56,8	88,4	379,6	г	
Залишок, коп	10	40	55	200	0	0	0	0	305		

З цієї таблиці видно, що внаслідок виконання процедури оптимізації отримано зазначену суму монет у 4153

коп, вага яких становить 379,6 г. Також залишиться ще декілька монет в сумі 305 коп. Задача розв'язана.

Приклад 2: задача про монети. Залишимо попередню постановку задачі (див. прикл. 1) з внесенням тільки деяких змін до формулювання самого завдання. Нове завдання полягатиме в тому, щоб відібрати зазначену суму монет, але вже з мінімальною їх загальною вагою, дотримуючись при цьому обмеження на наявну їх кількість.

Математична постановка задачі. Див. постановку задачі про монети (прикл. 1) з такими змінами. Необхідно під час відбору зазначеної суми монет певного номіналу мінімізувати загальну їхню вагу

$$F = \sum_{j=1}^n v_j x_j \rightarrow \min, \quad (3)$$

з дотриманням таких обмежень:

- на зазначену суму монет різного номіналу

$$\sum_{j=1}^n w_j x_j = S; \quad (4)$$

- на наявну кількість монет j -го номіналу $x_j \leq k_j, j = \overline{1, n}$,

Табл. 2.4. Результати розв'язання задачі про монети після процедури оптимізації

Номінал, коп	1	2	5	10	25	50	100*	100**	Сума		
Кількість, шт	1	1	0	15	22	27	8	13	Сума		
Сума, коп	1	2	0	150	550	1350	800	1300	4153	коп	
Сума, г	1,5	1,8	0	25,5	63,8	113,4	56,8	88,4	351,2	г	мін
Залишок, коп	11	44	90	160	0	0	0	0	305		

З цієї таблиці видно, що внаслідок виконання процедури оптимізації отримано зазначену суму монет у 4153 коп, вага яких становить вже 351,2 г, що на 28,4 г менше, ніж це показано в табл. 1.3. Також залишаться ще декілька монет різного номіналу в сумі 305 коп. Задача розв'язана. Звернемо увагу тільки на те, що в оптимальний розв'язок задачі зовсім не увійшли монети номіналом 5 коп, позаяк їх вага є значною порівняно з цінністю.

Отже, з двох постановок задач (прикл. 1 і 2) другій постановці можна надати більшу перевагу, позаяк вона дає змогу виконати не тільки замовлення на зазначену суму монет, але й з мінімальною їх вагою.

Приклад 3: задача про ранець. Задано набір предметів, для кожного з яких відомі: вага (0,41, 0,87, 1,0, 2,42, 3,58, 4,78, 5,93 і 7,63 кг); об'єм (2,0, 3,4, 1,9, 4,4, 2,0, 2,6, 1,4 і 5,1 дм³); кількість (3, 5, 9, 7, 3, 8, 3 і 6 шт). Також є ранець, в який можна вкладати предмети різного виду загальною вагою 32 кг. Завдання полягає в наповненні ранця зазначеної ваги, дотримуючись при цьому обмеження на наявну кількість предметів.

Математична постановка задачі. Введемо такі позначення:

- n – кількість видів предметів;
- $\tilde{V} = \{v_j, j = \overline{1, n}\}$ – вага предмету j -го виду, кг;
- $\tilde{W} = \{w_j, j = \overline{1, n}\}$ – об'єм предмету j -го виду, дм³;
- $\tilde{K} = \{k_j, j = \overline{1, n}\}$ – наявна кількість предметів j -го виду, шт;
- $\tilde{X} = \{x_j, j = \overline{1, n}\}$ – невідома кількість предметів j -го виду, шт;
- S – загальна вага ранця, кг.

Тоді задача має таке математичне формулювання: задано предмети n видів, кожен з яких має певну вагу

Табл. 3.1. Вхідні дані задачі про ранець

Характеристика предметів	Набір предметів j -го виду								Сума наявна		Набрати
	1	2	3	4	5	6	7	8			
Вага, кг	0,41	0,87	1,00	2,42	3,58	4,78	5,93	7,63			
Об'єм, дм ³	2,0	3,4	1,9	4,4	2,0	2,6	1,4	5,1			
Кількість, шт	3	5	9	7	3	8	3	6			
Сума, кг	1,23	4,35	9,00	16,94	10,74	38,24	17,79	45,78	144,07	кг	32
Сума, дм ³	6,0	17,0	17,1	30,8	6,0	20,8	4,2	30,6	132,5	дм ³	мін

а також за умови отримання натурального розв'язку задачі $x_j \in \mathbb{N}_0, j = \overline{1, n}$.

Наведена постановка задачі належить до варіанту задачі про обмежений ранець (англ. *Bounded Knapsack Problem*), де в ранець вказаної місткості потрібно укласти певну кількість предметів різного виду мінімальної ваги.

Методика розв'язання задачі. Складемо табл. 1.1, в яку внесемо значення всіх вхідних даних і розрахуємо ще деякі додаткові параметри. Аналогічно побудуємо табл. 1.2, в якій попередньо розрахуємо деякі допоміжні дані перед виконанням процедури оптимізації.

Для отримання зазначеної суми монет з мінімальною їх загальною вагою, дотримуючись при цьому обмеження на наявну їх кількість відповідного номіналу, використаємо певну процедуру оптимізації, внаслідок чого матимемо таку нову табл. 2.4.

(v_j), об'єм (w_j) та наявну кількість (k_j), які можна вкласти у ранець певної місткості. Необхідно укласти ранець зазначеної ваги предметами різного виду

$$\sum_{k=1}^n v_j x_j = S, \quad (5)$$

з дотриманням обмеження на наявну кількість предметів j -го виду $x_j \leq k_j, j = \overline{1, n}$, а також за умови отримання натурального розв'язку задачі $x_j \in \mathbb{N}_0, j = \overline{1, n}$.

Наведена постановка задачі належить до варіанту задачі про обмежений ранець (англ. *Bounded Knapsack Problem*), де в ранець вказаної місткості потрібно укласти певну кількість предметів різного виду.

Методика розв'язання задачі. Складемо табл. 3.1, в яку внесемо значення всіх вхідних даних і розрахуємо ще деякі додаткові параметри – сумарні вагу і об'єм.

Нехай для наповнення ранця зазначеної ваги відповідними предметами спочатку візьмемо по одному з них і розрахуємо табл. 3.2. З цієї таблиці видно, що внаслідок виконання таких дій отримаємо ранець вагою 26,62 кг (а потрібно 32 кг), об'єм якого становить 22,8 дм³.

Для отримання зазначеної ваги ранця, наповненого предметами різного виду, використаємо певну процедуру оптимізації, дотримуючись при цьому обмеження на наявну кількість предметів певного виду, внаслідок чого матимемо таку нову табл. 3.3. З цієї таблиці видно, що внаслідок виконання процедури оптимізації отримаємо зазначену вагу ранця в 32 кг, об'єм якого становить 28,8 дм³. Також залишаться ще деяка кількість не використаних предметів різного виду. Задача розв'язана.

Табл. 3.2. Розрахункові дані задачі про ранець перед процедурою оптимізації

Предмети	1	2	3	4	5	6	7	8	Сума		
Кількість, шт	1	1	1	1	1	1	1	1			
Сума, кг	0,41	0,87	1,00	2,42	3,58	4,78	5,93	7,63	26,62	кг	-5,38
Сума, дм ³	2,0	3,4	1,9	4,4	2,0	2,6	1,4	5,1	22,8	дм ³	
Залишок, шт	2	4	8	6	2	7	2	5	36	шт.	

Табл. 3.3. Результати розв'язання задачі про ранець після процедури оптимізації

Предмети	1	2	3	4	5	6	7	8	Сума		
Кількість, шт	3	1	0	1	1	5	0	0			
Сума, кг	1,23	0,87	0,00	2,42	3,58	23,90	0,00	0,00	32,00	кг	0,00
Сума, дм ³	6,0	3,4	0,0	4,4	2,0	13,0	0,0	0,0	28,8	дм ³	
Залишок, шт	0	4	9	6	2	3	3	6	33	шт.	

Приклад 4: задача про ранець. Залишимо попередню постановку задачі (див. прикл. 3) з внесенням тільки деяких змін до формулювання самого завдання. Нове завдання полягає в наповненні ранця зазначеної ваги предметами відповідного виду, об'єм яких має бути мінімальним, дотримуючись при цьому обмеження на наявну кількість предметів.

Математична постановка задачі. Див. постановку задачі (прикл. 3) з такими змінами. Необхідно під час наповнення ранця предметами різного виду мінімізувати їхній загальний об'єм

$$F = \sum_{j=1}^n w_j x_j \rightarrow \min, \quad (6)$$

з дотриманням таких обмежень:

- на зазначену вагу ранця предметами відповідного виду

$$\sum_{j=1}^n w_j x_j = S; \quad (7)$$

- на наявну кількість предметів j -го виду $x_j \leq k_j, j = \overline{1, n}$,

Табл. 4.3. Результати розв'язання задачі про ранець після процедури оптимізації

Предмети	1	2	3	4	5	6	7	8	Сума		
Кількість, шт	0	0	3	0	1	0	3	1			
Сума, кг	0,00	0,00	3,00	0,00	3,58	0,00	17,79	7,63	32,00	кг	0,00
Сума, дм ³	0,0	0,0	5,7	0,0	2,0	0,0	4,2	5,1	17,0	дм ³	мін
Залишок, шт	3	5	6	7	2	8	0	5	36	шт.	

З цієї таблиці видно, що внаслідок виконання процедури оптимізації отримано зазначену вагу ранця 32 кг, об'єм якого становитиме всього 17,0 дм³, що на 5,8 дм³ менше, ніж у табл. 3.3. Також залишиться ще деяка кількість не використаних предметів різного виду. Задача розв'язана.

Отже, з двох постановок задач (прикл. 3 і 4) другій постановці можна надати більшу перевагу, позаяк вона дає змогу не тільки укласти ранець предметами різного виду зазначеної ваги, але й мінімізувати їхній загальний об'єм.

Приклад 5: задача про декілька ранців. Задано набір предметів, для кожного з яких відомі: вага (0,41, 0,87, 1,0, 2,42, 3,58, 4,78, 5,93 і 7,63 кг); об'єм (2,0, 3,4, 1,9, 4,4, 2,0, 2,6, 1,4 і 5,1 дм³); кількість (3, 5, 9, 7, 3, 8, 3 і 6 шт). Також є два ранці, в які можна вкладати предмети різного виду загальною вагою не менше 25 і 32 кг відповідно. Завдання полягає в тому, щоб укласти два ранці зазначеної ваги, дотримуючись при цьому обмеження на наявну кількість предметів.

Математична постановка задачі. Введемо такі позначення:

- n – кількість видів предметів;
- m – кількість ранців різної ваги;
- $\tilde{V} = \{v_j, j = \overline{1, n}\}$ – вага предмету j -го виду, кг;
- $\tilde{W} = \{w_j, j = \overline{1, n}\}$ – об'єм предмету j -го виду, дм³;

а також за умови отримання натурального розв'язку задачі $x_j \in \mathbb{N}_0, j = \overline{1, n}$.

Наведена постановка задачі належить до варіанту задачі про обмежений ранець (англ. *Bounded Knapsack Problem*), де в ранець вказаної місткості потрібно укласти певну кількість предметів різного виду мінімального об'єму.

Методика розв'язання задачі. Складемо табл. 3.1, в яку внесемо значення всіх вхідних даних і розрахуємо відповідні додаткові параметри. Аналогічно побудуємо табл. 3.2, в якій попередньо розрахуємо деякі допоміжні дані для розв'язання задачі про ранець перед виконанням процедури оптимізації.

Для отримання зазначеної ваги ранця, наповненого предметами різного виду, використаємо певну процедуру оптимізації на їхній об'єм, дотримуючись при цьому обмеження на наявну кількість предметів певного виду, внаслідок чого матимемо таку нову табл. 4.3.

- $\tilde{K} = \{k_j, j = \overline{1, n}\}$ – наявна кількість предметів j -го виду, шт;
- $\tilde{X} = \{\tilde{X}_i = \{x_{ij}, j = \overline{1, n}\}, i = \overline{1, m}\}$ – невідома кількість предметів j -го виду, які потрібно помістити в i -ий ранець, шт;
- $\tilde{S} = \{s_i, i = \overline{1, m}\}$ – загальна вага i -го ранця, кг.

Тоді задача має таке математичне формулювання: задано предмети n видів, кожен з яких має певну вагу (v_j), об'єм (w_j) та наявну кількість (k_j). Також відома загальна вага кожного ранця (s_i), в які можна помістити наявні предмети певного виду. Необхідно укласти кожен ранець зазначеної ваги предметами різного виду

$$\sum_{j=1}^n v_j x_{ij} \geq s_i, i = \overline{1, m}, \quad (8)$$

з дотриманням обмеження на наявну кількість предметів j -го виду

$$\sum_{i=1}^m x_{ij} \leq k_j, j = \overline{1, n}, \quad (9)$$

а також за умови отримання натурального розв'язку задачі $x_{ij} \in \mathbb{N}_0, j = \overline{1, n}; i = \overline{1, m}$.

Наведена постановка задачі про декілька ранців належить до варіанту задачі про множинний ранець (англ. *Multiple Knapsack Problem*), де в декілька ранців, кожен з яких має свою місткість, потрібно укласти певну кількість предметів різного виду.

Методика розв'язання задачі. Складемо таблицю, в яку внесемо значення всіх вхідних даних і розрахуємо ще деякі додаткові параметри – сумарні вагу і об'єм.

Нехай для наповнення ранців зазначеної ваги відповідними предметами спочатку візьмемо по одному з них і розрахуємо табл. 5.2. З цієї таблиці видно, що внаслідок виконання таких дій отримаємо два ранці вагою по 26,62 кг, об'єм кожного з яких становить всього

22,8 дм³. При цьому вага 1-го ранця перевищує на 1,62 кг, водночас в 2-му ранці не вистачає 5,38 кг, а їхній загальний об'єм становить 45,6 дм³.

Для отримання зазначеної ваги кожного з двох ранців, наповнених предметами різного виду, використаємо певну процедуру оптимізації, дотримуючись при цьому обмеження на наявну кількість предметів певного виду, внаслідок чого матимемо таку нову табл. 5.3.

Табл. 5.1. Вхідні дані задачі про два ранці

Характеристика предметів	Набір предметів j -го виду								Сума наявна		Набрати
	1	2	3	4	5	6	7	8			
Вага, кг	0,41	0,87	1,00	2,42	3,58	4,78	5,93	7,63			
Об'єм, дм ³	2,0	3,4	1,9	4,4	2,0	2,6	1,4	5,1			
Кількість, шт	3	5	9	7	3	8	3	6		кг	25
Сума, кг	1,23	4,35	9,00	16,94	10,74	38,24	17,79	45,78	144,07	кг	32
Сума, дм ³	6,0	17,0	17,1	30,8	6,0	20,8	4,2	30,6	132,5	дм ³	мін

Табл. 5.2. Розрахункові дані задачі про два ранці перед процедурою оптимізації

Предмети	1	2	3	4	5	6	7	8	Сума		
1. Кількість, шт	1	1	1	1	1	1	1	1		кг	25
2. Кількість, шт	1	1	1	1	1	1	1	1		кг	32
Сума, шт	2	2	2	2	2	2	2	2			
1 ранець											
Сума, кг	0,41	0,87	1,00	2,42	3,58	4,78	5,93	7,63	26,62	кг	1,62
Сума, дм ³	2,0	3,4	1,9	4,4	2,0	2,6	1,4	5,1	22,8	дм ³	
2 ранець											
Сума, кг	0,41	0,87	1,00	2,42	3,58	4,78	5,93	7,63	26,62	кг	-5,38
Сума, дм ³	2,0	3,4	1,9	4,4	2,0	2,6	1,4	5,1	22,8	дм ³	
Залишок, шт	1	3	7	5	1	6	1	4	28	шт.	7,00
											45,6

Табл. 5.3. Результати розв'язання задачі про два ранці після процедури оптимізації

Предмети	1	2	3	4	5	6	7	8	Сума		
1. Кількість, шт	1	0	2	1	1	1	2	0		кг	25
2. Кількість, шт	0	2	3	2	2	0	0	2		кг	32
Сума, шт	1	2	5	3	3	1	2	2			
1 ранець											
Сума, кг	0,41	0,00	2,00	2,42	3,58	4,78	11,86	0,00	25,05	кг	0,05
Сума, дм ³	2,0	0,0	3,8	4,4	2,0	2,6	2,8	0,0	17,6	дм ³	
2 ранець											
Сума, кг	0,00	1,74	3,00	4,84	7,16	0,00	0,00	15,26	32,00	кг	0,00
Сума, дм ³	0,0	6,8	5,7	8,8	4,0	0,0	0,0	10,2	35,5	дм ³	
Залишок, шт	2	3	4	4	0	7	1	4	25	шт.	0,05
											53,1

З цієї таблиці видно, що внаслідок виконання процедури оптимізації отримано зазначену вагу 1-го ранця 25,05 кг (що на 0,05 кг більше від потрібної), а вагу 2-го ранця – 32,00 кг. Водночас об'єм 1-го ранця має 17,6 дм³, а об'єм 2-го ранця – 35,5 дм³, що в сумі становить 53,1 дм³. Також залишиться ще деяка кількість не використаних предметів різного виду. Задача розв'язана.

Приклад 6: задача про декілька ранців. Залишимо попередню постановку задачі (див. прикл. 5) з внесенням тільки деяких змін до формулювання самого завдання. Нове завдання полягає в наповненні двох ранців зазначеної ваги предметами різного виду, загальний об'єм яких має бути мінімальним, дотримуючись при цьому обмеження на наявну кількість предметів.

Математична постановка задачі. Див. постановку задачі (прикл. 5) з такими змінами. Необхідно під час наповнення двох ранців предметами різного виду мінімізувати загальний їхній об'єм

$$F = \sum_{i=1}^m \sum_{j=1}^n v_j x_{ij} \rightarrow \min, \quad (10)$$

з дотриманням таких обмежень:

- на зазначену вагу i -го ранця

$$\sum_{j=1}^n w_j x_{ij} \geq s_i, \quad i = \overline{1, m}; \quad (11)$$

- на наявну кількість предметів j -го виду

$$\sum_{i=1}^m x_{ij} \leq k_j, \quad j = \overline{1, n}, \quad (12)$$

а також за умови отримання натурального розв'язку задачі $x_{ij} \in \mathbb{N}_0, j = \overline{1, n}; i = \overline{1, m}$.

Наведена постановка задачі про декілька ранців належить до варіанту задачі про *множинний ранець* (англ. *Multiple Knapsack Problem*), де в декілька ранців, кожен зі своєю вказаною місткістю, потрібно укласти певну кількість предметів різного виду мінімального об'єму.

Методика розв'язання задачі. Складемо табл. 5.1, в яку внесемо значення всіх вхідних даних і розрахуємо ще деякі додаткові параметри. Аналогічно побудуємо табл. 5.2, в якій попередньо розрахуємо деякі допоміжні дані для розв'язання задачі про два ранці перед виконанням процедури оптимізації.

Для отримання зазначеної ваги двох ранців, наповнених предметами різного виду, використаємо певну процедуру оптимізації на їхній мінімальний об'єм, дотримуючись при цьому обмеження на зазначену вагу та наявну кількість предметів певного виду, внаслідок чого матимемо таку нову табл. 6.3.

З цієї таблиці видно, що внаслідок виконання процедури оптимізації отримано зазначену вагу 1-го ранця 25,08 кг (що на 0,08 кг більше від потрібної), а вагу 2-го ранця – 32,13 кг (що на 0,13 кг більше від потрібної).

Водночас об'єм 1-го ранця має 13,8 дм³, а об'єм 2-го ранця – 12,0 дм³, що в сумі становить всього 25,8 дм³, а це на 27,3 дм³ менше, ніж у табл. 5.3. Також залишить-

ся ще деяка кількість не використаних предметів різного виду. Задача розв'язана.

Табл. 6.3. Результати розв'язання задачі про два рюкзаки після процедури оптимізації

Предмети	1	2	3	4	5	6	7	8	Сума		
1. Кількість, шт	0	0	0	0	3	3	0	0		кг	25
2. Кількість, шт	0	0	0	0	0	3	3	0		кг	32
Сума, шт	0	0	0	0	3	6	3	0			
1 рюнець											
Сума, кг	0,00	0,00	0,00	0,00	10,74	14,34	0,00	0,00	25,08	кг	0,08
Сума, дм ³	0,0	0,0	0,0	0,0	6,0	7,8	0,0	0,0	13,8	дм ³	
2 рюнець											
Сума, кг	0,00	0,00	0,00	0,00	0,00	14,34	17,79	0,00	32,13	кг	0,13
Сума, дм ³	0,0	0,0	0,0	0,0	0,0	7,8	4,2	0,0	12,0	дм ³	
Залишок, шт	3	5	9	7	0	2	0	6	32	шт.	0,21
											25,8

Отже, з двох постановок задач (прикл. 5 і 6) другій постановці можна надати більшу перевагу, позаяк вона дає змогу не тільки укласти два рюкзаки предметами різного виду зазначеної ваги, але й мінімізувати їхній загальний об'єм у кожному з рюнків.

Для реалізації будь-якої з наведених вище постановок задач розроблено відповідне програмне забезпечення (рис. 4), функціонал якого дає змогу занести вхідні дані залежно від кількості рюнків, наявних предметів різних видів, їхньої кількості, ваги та цінності, а також виконати відповідні розрахунки. В програмі використано жадібний алгоритм для отримання оптимального розв'язку задачі.

MyBackPack СТАТТІ МАРШРУТИ СПОРЯДЖЕННЯ ЗАДАЧІ ПРО НАС ПРОФІЛЬ ВИХІД

Задача про декілька рюнків

Вхідні дані

Кількість рюнків:

Місіть рюнка 1, кг:

Місіть рюнка 2, кг:

Набір предметів	1	2	3	4	5	6	7	8	Найважна сума
Вага, кг	0,41	0,87	1,0	2,42	3,58	4,78	5,93	7,63	
Об'єм, дм ³	2,0	3,4	1,9	4,4	2,0	2,6	1,4	5,1	
Кількість, шт	3	5	9	7	3	8	3	6	
Сума, кг	1,23	4,35	9,0	16,94	10,74	38,24	17,79	45,78	144,07
Сума, дм ³	6,0	17,0	17,1	30,8	6,0	20,8	4,2	30,6	132,5

Результати розрахунку

Набір предметів	1	2	3	4	5	6	7	8	Найважна сума
1 рюнець									
Кількість, шт	0	0	0	0	3	3	0	0	6
Сума, кг	0,0	0,0	0,0	0,0	10,76	14,34	0,0	0,0	25,08
Сума, дм ³	0,0	0,0	0,0	0,0	6,0	7,8	0,0	0,0	13,8
2 рюнець									
Кількість, шт	0	0	0	0	0	3	3	0	6
Сума, кг	0,0	0,0	0,0	0,0	0,0	14,34	17,79	0,0	32,13
Сума, дм ³	0,0	0,0	0,0	0,0	0,0	7,8	4,2	0,0	12,0
Кількість загальна, шт	0	0	0	0	3	6	3	0	12
Залишок, кг	3	5	9	7	0	2	0	6	32

Рис. 4. Форма для розв'язання задачі про рюнець

Висновки. Розроблено ефективну методику розв'язання практичних постановок задачі про рюнець, яка дає змогу отримати адекватні результати розрахунку, провести їх змістовний аналіз та вибрати вдалі постановки для подальшого застосування. За результатами проведеного дослідження можна зробити такі основні висновки.

1. З'ясовано, що процедура укладання туристичного наплічника зазвичай є проблемою як для початківців, так і бувалих мандрівників. Водночас, досвідчені туристи в таких ситуаціях володіють деякими загальними правилами, які дають їм змогу вкладати найбільш потрібні речі не тільки встановленої місткості та мінімальної ваги, але й дотримуватись деякого порядку розмі-

щення цих речей в наплічнику і надати йому традиційну форму, що забезпечує зручність тривалого його перенесення.

2. Виявлено, що класична постановка задачі про рюнець належить до задач цілочисельного програмування, проаналізовано можливі варіанти її постановок, з'ясовано основні причини її широкого застосування в різних областях знань: в математиці, інформатиці та на стику цих наук – в криптографії.

3. Встановлено, що задача про рюнець належить до класу NP-повних задач комбінаторної оптимізації, тому для неї немає поліноміального алгоритму, здатного її розв'язати за розумний проміжок часу. Визначено особливості застосування точних методів розв'язання задачі про рюнець, проаналізовано метод повного перебору можливих варіантів, метод гілок і меж, жадібний алгоритм і методи динамічного програмування. Наведено рекомендації щодо вибору серед них найпридатнішого для розв'язання запропонованих у роботі постановок задач.

4. Наведено приклади деяких практичних постановок задачі про рюнець, здійснено їхню формалізацію та алгоритмізацію, запропоновано адекватний метод розв'язання, а також проведено змістовний аналіз отриманих результатів розрахунку, на підставі яких вибрано вдалі постановки задач для подальшого їх застосування.

5. Зроблено відповідні висновки та надано практичні рекомендації щодо використання розробленого ПЗ для розв'язання різних постановок задач про укладання як туристичного рюнка, так і їх інших варіацій.

Перелік використаних джерел

- Andonov, Rumen; Poirriez, Vincent; Rajopadhye, Sanjay. (2000). "Unbounded Knapsack Problem : dynamic programming revisited". *European Journal of Operational Research*, 123(2), 168–181.
- Batishhev, D. I., Neimark, E. A., & Starostin, N. V. (2007). *Применение генетических алгоритмов к решению задач дискретной оптимизации*. Nizhny Novgorod. [In Russian].
- Brethauer, K. M., & Shetty, B. (2002). The nonlinear Knapsack Problem – algorithms and applications. *European Journal of Operational Research – Elsevier*, Vol. 138, Iss. 3, 459–472. [https://doi.org/10.1016/S0377-2217\(01\)00179-5](https://doi.org/10.1016/S0377-2217(01)00179-5)
- Burkov, V. N., Gorgidze, I. A. (Ed.), & Lovetckii, S. E. (1974). *Priladnye zadachi teorii grafov*. Tbilisi: Vychislitelnyi tsentr AN SSSR, 231 p. [In Russian].
- Dantzig, G. B. (1957). Discrete-Variable Extremum Problems. *Operations Research*, Vol. 5, Iss. 2, 266–277. Institute for Operations Research and the Management Sciences. <https://doi.org/10.1287/OPRE.5.2.266>
- Diffie, Whitfield, & Hellman, Martin E. (1977). Exhaustive Cryptanalysis of the NBS Data Encryption Standard. *Journal Computer*, Vol. 10, pp. 74–84. June 1977.

- Diubin, G. N., & Korbut, A. A. (1999). Zhadnye algoritmy dlia zadachi o rance: povedenie v srednem. *Sibirskii zhurnal industrialnoi matematiki*, Vol. II, № 2(4), pp. 68–93. [In Russian].
- Dorigo, M. (1992). Optimization, Learning and Natural Algorithms. *PhD Thesis. Dipartimento di Elettronica*, Politecnico Di Milano. Italy, 140 p.
- Gabidulin, E. M., Kshevetckii, A. S., & Kolybelnikov, A. I. (2011). Zashhita informatsii: uchebnoe posobie. Moskva: MFTI, 225 p. [In Russian].
- Gallo, G., Hammer, P. L., & Simeone, B. (2009). Quadratic Knapsack Problems. *Mathematical Programming Studies*, (Vol. 12, pp. 132–149). February 24. <https://doi.org/10.1007/BFb0120892>
- Gallo, G.; Simeone, B. (1988). "On the Supermodular Knapsack Problem". *Mathematical Programming Studies*, 45(1–3): 295–309. <https://doi.org/10.1007/bf01589108>
- Grusho, A. A., Primenko, E. A., & Timonina, E. E. (2000). *Analiz i sintez kriptotalgoritmov*. Kurs lektcii. Moskva: Nauka, 110 p. [In Russian].
- Iukhimenko, B. I. (2004). Uskorenyi algoritm metoda vetvei i granic dlia resheniia zadachi tselochislennogo lineinogo programmirovaniia. *Trudy Odesskogo politekhnicheskogo un-ta*, 2, 223–226. [In Russian].
- Iukhimenko, B. I., & Kozina, Iu. Iu. (2005). Sravnitelnaia kharakteristika algoritmov metoda vetvei i granic dlia resheniia zadach tselochislennogo lineinogo programmirovaniia. *Trudy Odesskogo politekhnicheskogo un-ta*, 2, 199–204. [In Russian].
- Karp, R. M. (1972). Reducibility Among Combinatorial Problems. In: *R. E. Miller and J. W. Thatcher (Ed.). Complexity of Computer Computations*, (pp. 85–103). New York: Plenum.
- Karte, B. (1986). Kombinatorische Optimierung und algorithmische Prinzipien. *Okonomische Prognose, Entscheidungs und Gleichgewichtsmodelle*, (pp. 286–341). Weinheim: VCH Verlagsgesellschaft.
- Kellerer, H., Pferschy, U., & Pisinger, D. (2004). *Knapsack Problems – Springer Science + Business Media*, 548 p. <https://doi.org/10.1007/978-3-540-24777-7>
- Kin Ming Lai. (2001). Knapsack Cryptosystems: The Past and the Future. [In Russian].
- Knapsack problem. (2019). From Wikipedia, the free encyclopedia. Retrieved from: https://en.wikipedia.org/wiki/Knapsack_problem
- Koblitz, N. (2001). Kurs teorii chisel v kriptografii. Izd. 2-oe. Moskva: Nauchnoe izdatelstvo TVP, 254. [In Russian].
- Kulik, A., & Shachnai, H. (2010). There is no EPTAS for two-dimensional knapsack. *Information Processing Letters*, 110(16), 707–710. <https://doi.org/10.1016/j.ipl.2010.05.031>.
- Levitin, A. V. (2006). Algoritmy. Vvedenie v razrabotku i analiz. Moskva: Viliams, 576 p. [In Russian].
- Martello, S., & Toth, P. (1990). Knapsack Problems: algorithms and computer implementations. *John Wiley & Sons Ltd.*, 29, 50, 296 p.
- Mathews, G. B. (1897). On the Partition of numbers. *Proceedings of the London Mathematical Society*, Vol. s1-28, Issue 1, November 1896, pp. 486–490, <https://doi.org/10.1112/plms/s1-28.1.486>.
- Merkle Ralph C., Hellman Martin E. (1981). On the security of multiple encryption. *Journal Communications of the ACM*, num. 7 Vol. 24, pp. 465–467. July 1981
- Okulov, S. (2007). Programirovanie v algoritmkh. Izd. 1-oe, Binom. Laboratoriia znani, 384. [In Russian].
- Osipian, V. O. (2006). O sisteme zashhity informatsii na osnove problema riukzaka. *Izvestiia Tomskogo politekhnicheskogo universiteta*, Vol. 309, № 2, 209–212. [In Russian].
- Riedhammer, K., Gillick, D., Favre, B., & Hakkani-Tür, D. (2008). Packing the Meeting Summarization Knapsack. In: *Proceedings of the 9th International Conference of the ISCA (Interspeech 2008)*, (pp. 2434–2437). Brisbane Australia: Proc. Interspeech.
- Romanovskii, I. V. (1977). Algoritmy resheniia ekstremalnykh zadach. Moskva: Nauka, 352 p. [In Russian].
- Salomaa, A. (1995). Kriptografiia s otkrytym kliuchom = Public-Key Cryptography. Moskva: Mir, 102–150. [In Russian].
- Shnaier, B. (2002). Prikladnaia kriptografiia. Protokoly, algoritmy, iskhodnye teksty na iazyke Si = Applied Cryptography. Protocols, Algorithms, and Source Code in C. Izd. 2-oe. Moskva: Triumf, 816 p. [In Russian].
- Sigal, I. Kh., & Ivanova, A. P. (2007). Vvedenie v prikladnoe diskretnoe programmirovaniie. Moskva: Fizmatlit, 304. [In Russian].
- Skiena, S. S. (September 1999). "Who is Interested in Algorithms and Why? Lessons from the Stony Brook Algorithm Repository". *ACM SIGACT News*. 30 (3): 65–74. CiteSeerX 10.1.1.41.8357
- Stiven S. Skiena. (2011). Algoritmy. Rukovodstvo po razrabotke. Izd. 2-e, SPb.: BKhV-Peterburg, 720 p. [In Russian].

Yu. I. Hrytsiuk, O. A. Nemova

Lviv Polytechnic National University, Lviv, Ukraine

FORMALIZATION OF TASK STATEMENTS FOR A KNAPSACK PROBLEM AND ALGORITHMS TO SOLVE THEM

The paper describes the formalization of task statements for a tourist knapsack problem and the efficient algorithms to solve them, which generally allowed obtaining adequate results of calculation, conducting a meaningful analysis, and selecting the successful task statements for their further application. In the course of research we have revealed that the knapsack problem usually causes difficulties for both beginners and experienced travellers. At the same time, experienced tourists in such situations have some general rules that enable them to put the most necessary items not only in determined volume and minimum weight, but also to adhere to the order of placement of these items in the knapsack and give it a traditional form that provides convenient long-term carrying. We have also found that the classical task statement of the knapsack problem belongs to the problems of integer programming; it allows a large number of different generalizations, depending on the restrictions imposed on the knapsack, on the items or their choice, and also on the condition for obtaining the optimal solution of the problem, i.e. Boolean or quantitative. The possible variants of its statements are analyzed; the main reasons of wide application in different fields of knowledge are studied. Furthermore, the knapsack problem is defined to belong to the class of NP-complete combinatorial optimization problems. Therefore, there is no polynomial algorithm capable of solving it for a reasonable time interval. The peculiarities of the application of exact methods for solving the knapsack problem are determined. The method of full overview of possible variants, the branch and bound method, the greedy algorithm, and the methods of dynamic programming are analyzed as well. Some recommendations on choosing among them the method that is the most suitable for solving tasks proposed in the work are given. Some examples of the practical task statements for the knapsack problem are given, and their formalization, algorithmization and program realization are also carried out. An adequate method for their solution has been proposed, and a meaningful analysis of the obtained calculation results has been performed. Thus, successful task statements for their further application are set on based of the analysis.

Keywords: the knapsack problem; an integer programming problem; combinatorial optimization problems; restrictions and conditions of the problem; the condition of the optimal solution of the problem.