

## МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

### METHODICAL MATERIALS

*Вестник Сыктывкарского университета.*

*Серия 1: Математика. Механика. Информатика. 2024.*

*Выпуск 1 (50)*

*Bulletin of Syktovkar University.*

*Series 1: Mathematics. Mechanics. Informatics. 2024; 1 (50)*

Научная статья

УДК 519.1, 378

[https://doi.org/10.34130/1992-2752\\_2024\\_1\\_55](https://doi.org/10.34130/1992-2752_2024_1_55)

### КОМПЬЮТЕРНЫЕ ИГРЫ И КОМБИНАТОРНЫЕ ЗАДАЧИ

Надежда Николаевна Бабикина<sup>1</sup>,

Надежда Олеговна Котелина<sup>1</sup>,

Мария Александровна Валужева<sup>2</sup>,

Николай Андреевич Старцев<sup>3</sup>

<sup>1</sup>Сыктывкарский государственный университет  
имени Питирима Сорокина

<sup>2</sup>Санкт-Петербургский государственный институт культуры

<sup>3</sup>Санкт-Петербургский государственный электротехнический  
университет «ЛЭТИ» имени В. И. Ульянова (Ленина)

**Аннотация.** В статье предлагаются для обсуждения авторские задачи по комбинаторике, составленные по мотивам компьютерных игр. Представлены примеры задач для практических занятий и лабораторных работ по темам: правила суммы и произведения, перестановки с повторениями и без, размещения с повторениями и без, сочетания с повторениями и без, разбиение числа на слагаемые и разбиение числа на слагаемые, каждое из которых не превосходит заданного значения, формула включений и исключений. Задачи апробированы в процессе обучения дисциплине «Дискретная математика» студентов направления подготовки «Прикладная информатика».

**Ключевые слова:** комбинаторика, комбинаторные задачи, компьютерные игры, обучение, авторские задачи

**Для цитирования:** Бабилова Н. Н., Котелина Н. О., Валуева М. А., Старцев Н. А. Компьютерные игры и комбинаторные задачи // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика*. 2024. Вып. 1 (50). С. 55–72. [https://doi.org/10.34130/1992-2752\\_2024\\_1\\_55](https://doi.org/10.34130/1992-2752_2024_1_55)

Article

### Computer games and combinatorial problems

Nadezhda N. Babikova<sup>1</sup>, Nadezhda O. Kotelina<sup>1</sup>,  
Marija A. Valueva<sup>2</sup>, Nikolaj A. Startsev<sup>3</sup>

<sup>1</sup>Pitirim Sorokin Syktyvkar State University, valmasha@mail.ru

<sup>2</sup>St. Petersburg State University of Culture,

<sup>3</sup>St. Petersburg State Electrotechnical University "LETI"  
named after V. I. Ulyanov (Lenin)

**Abstract.** The author's problems in combinatorics, based on computer games, are offered for discussion in the article. Examples of problems for practical classes and laboratory work are presented on the following topics: sum and product rules, permutations with and without repetitions, placements with and without repetitions, combinations with and without repetitions, partitioning a number into terms and partitioning a number into terms, each of which does not exceed given value, inclusion and exclusion formula. The problems were tested in the process of teaching the discipline "Discrete Mathematics" to students in the field of study "Applied Informatics".

**Keywords:** combinatorics, combinatorial problems, computer games, training, author's problems

**For citation:** Babikova N. N., Kotelina N. O., Valueva M. A., Startsev N. A. Computer games and combinatorial problems. *Vestnik Syktyvkarского университета. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University, Series 1: Mathematics. Mechanics. Informatics], 2024, no 1 (50), pp. 55–72. (In Russ.) [https://doi.org/10.34130/1992-2752\\_2024\\_1\\_55](https://doi.org/10.34130/1992-2752_2024_1_55)

### Введение

Одной из основных целей обучения математике в вузе является формирование способности студентов применять предметные знания вне предмета – как при изучении других дисциплин, так и в последующей

профессиональной деятельности, т. е. способности осуществлять *трансфер* (перенос) *знаний* в новый контекст.

Анализ педагогических и психологических исследований проблемы трансфера математических знаний позволяет выделить два основных условия, необходимых для формирования способности к применению знаний в новом контексте: высокий уровень усвоения предметных знаний (именно высокий, а не средний) [1, с. 18] и контекстная насыщенность прикладных задач [2, с. 18]. Разнообразие контекстов в учебных прикладных задачах, внешне различных и неоднородных, повышает вероятность успешного трансфера знаний в будущем [3, с. 316]. Желание разнообразить контекст задач по комбинаторике легло в основу исследования, результаты которого представлены в статье.

История становления и развития комбинаторики как науки тесно связана с играми и головоломками. В мире комбинаторных задач подбрасывают монетки и бросают кости, играют в домино и шахматы, выигрывают в лотерею и проигрывают в карты. Компьютерные игры стали неотъемлемой частью жизни современного человека, и их включение в мир традиционных комбинаторных задач представляется естественным и логичным.

## 1. Материалы и методы

На протяжении нескольких лет Н. Н. Бабиковой в процессе обучения дисциплине «Дискретная математика» (направление подготовки «Прикладная информатика», СГУ им. Питирима Сорокина) в рамках самостоятельной работы студентов по разделу «Комбинаторика» используется задание: «Составьте задачу(и) на комбинаторные конфигурации на основе любой компьютерной игры и решите ее (их)». На основе этого опыта авторами было решено разработать задачи по мотивам компьютерных игр на общие правила и основные комбинаторные объекты: правила суммы и произведения, расчет при условии «хотя бы один», перестановки с повторениями и без, размещения с повторениями и без, сочетания с повторениями и без, разбиение числа на слагаемые и разбиение числа на слагаемые, каждое из которых не превосходит заданного значения, формула включений и исключений.

Два автора — преподаватели «Дискретной математики», игровой опыт которых не слишком значителен (любимая игра — «Растения против зомби»). Два автора — студенты, изучавшие в вузе «Дискретную математику» и имеющие большой и разнообразный игровой опыт. Со-

ставленные авторами задачи апробированы в процессе обучения дисциплине «Дискретная математика» (направление подготовки «Прикладная информатика»). Задачи использовались на практических и лабораторных занятиях (реализация алгоритмов на языке Python), в качестве примеров в лекционном материале.

## 2. Результаты исследования

Рассмотрим примеры задач, различных по уровню сложности.

### Игры «Собери / найди пары»

Имеется 9 пар одинаковых карточек (рис. 1). Сколькими способами можно расположить карточки на экране?



Рис. 1. Яндекс игры: Найди пару – большой выпуск

Не имеет значения, как карточки расположены: будут ли они выстроены в один ряд, расположены в форме прямоугольника или просто случайно разбросаны по экрану. Важно лишь то, что они упорядочены. Это задача на **перестановки с повторениями**. Число различных вариантов расположения карточек —  $P(2, 2, 2, 2, 2, 2, 2, 2, 2) = 18!/2^9 = 12\,504\,636\,144\,000$ .

Иногда в подобных играх требуется собирать не пары, а тройки одинаковых карточек. Если имеется 6 троек одинаковых карточек, то количество вариантов их расположения составит  $P(3, 3, 3, 3, 3, 3) = 18!/6^6$ .

### Игра «Растения против зомби»

В игре «Растения против зомби» игроку нужно выбирать и выращивать растения, чтобы защитить свой дом от нападения орды зомби. По мере продвижения в игре число ячеек для выбора растений и число видов растений увеличивается (рис. 2). При этом растения можно

сгруппировать по функциям: есть растения для защиты, для дальнего боя, для ближнего боя и другие.



Рис. 2. Каталог растений в игре «Растения против зомби»

На этапе «Бассейн» (рис. 3) у игрока имеется 7 ячеек для выбора растений из 14 доступных видов. Растения по функциям делятся следующим образом: генерация света — подсолнух; защита от нападения — стенорех, картофельная мина, колючка; оборона на воде — лист кувшинки и водоросли, нанесение большого урона — вишневая бомба и перец; ближний бой — зубастик и кабачок; дальний бой — горохострел, огорошиватель, тройной горохострел, ледяной горохострел.



Рис. 3. Выбор растений на этапе «Бассейн»

Обязательно нужно выбрать подсолнух для генерации света и лист кувшинки, на который в бассейне устанавливаются все остальные рас-

тения. Игровой опыт показывает, что желательно иметь: хотя бы по одному растению для защиты от нападений и для нанесения большого урона, хотя бы два растения для дальнего боя. Сколькими способами можно выбрать растения?

Сначала выберем обязательные и желательные растения. Выбираем подсолнух и лист кувшинки. Выбрать любое растение для защиты от нападений можно  $C_3^1$  способами, любое растение для нанесения большого урона —  $C_2^1$  способами, два растения для дальнего боя —  $C_4^2$  способами (**сочетания без повторений**). По **правилу произведения** получаем  $C_3^1 \cdot C_2^1 \cdot C_4^2 = 18$  способов. После этого у нас остается: два растения для защиты от нападений, одно растение для нанесения большого урона, два растения для дальнего боя, два растения для ближнего боя и водоросли. Выбрать нам нужно еще одно растение, по **правилу суммы** это можно сделать  $(2 + 1 + 2 + 2 + 1) = 8$  способами. По правилу произведения общее число вариантов выбора  $18 \cdot 8 = 144$ .

### Игры «Dota 2» и «League of Legends»

Игры «Dota 2» и «League of Legends» — одни из самых популярных многопользовательских онлайн-игр в мире. В каждом матче обеих игр две команды по пять игроков сталкиваются на симметричной карте. Структура игровых карт одинакова, они состоят из трёх линий: верхней (top), центральной (mid) и нижней (bot), а также участков леса (рис. 4). Матч начинается с выбора героев игроками каждой команды, но правила выбора героев в League of Legends и Dota 2 различаются.



Рис. 4. Карты игр «League of Legends» (слева) и «Dota 2» (справа)

В Dota 2 на сегодняшний день 124 различных героя, каждый герой может быть выбран только один раз. Сколькими способами можно собрать каждую команду, если герои выбираются командами по очереди?

Для команды, выбирающей первой, по правилу произведения имеется  $124 \cdot 122 \cdot 120 \cdot 118 \cdot 116 = 24\,848\,647\,680$  способов. Для второй команды —  $123 \cdot 121 \cdot 119 \cdot 117 \cdot 115 = 23\,829\,841\,035$ .

В начальной стадии игры развитие происходит на трёх линиях, надо распределить пять героев команды по линиям так, чтобы на каждой линии оказался **хотя бы один** герой.

Распределить 5 героев по 3 линиям можно  $3^5$  способами (**размещения с повторениями**). При этом в  $3 \cdot 2^5$  случаях одна из линий окажется свободной, а в 3 случаях свободными окажутся две линии. Тогда по **формуле включений и исключений** получаем, что распределить 5 героев по 3 линиям так, чтобы на каждой линии оказался хотя бы один герой, можно  $3^5 - 3 \cdot 2^5 + 3 = 150$  способами. Эта задача относится к общей схеме «раскладка  $n$  различных предметов по  $m$  различным ящикам так, чтобы в каждом ящике лежал хотя бы один предмет» [4, с. 102].

В Dota 2 у каждого героя есть своя зона обзора, в рамках которой он может видеть все, что происходит вокруг. Остальная часть карты покрыта туманом войны. Лучший способ увидеть сквозь туман — это поставить варды. Вард — предмет, который позволяет видеть в тумане. Варды бывают двух типов: *observer* — дает обзор, *sentry* — позволяет видеть невидимых существ. В самом начале игры команде доступно два *observer* варда и три *sentry* варда. Сколькими способами можно распределить все варды между пятью героями команды?

Сначала посчитаем, сколькими способами можно распределить 2 *observer* варда между 5 героями. Если 2 варда распределяются между разными героями, то число вариантов соответствует числу сочетаний без повторений  $C_5^2 = 10$ . И есть 5 способов распределения обоих вардов одному из героев. Таким образом, общее количество способов распределить 2 *observer* варда равно  $10 + 5 = 15$ .

При распределении 3 *sentry* вардов среди 5 героев возможны три ситуации. По одному варду достается трём разным героям из пяти —  $C_5^3 = 10$  вариантов выбора героев. Если варды получают два героя, то сначала выберем одного героя и отдадим ему два варда — 5 способов, а потом выберем второго героя из оставшихся четырех и отдадим ему один вард — 4 способа, всего 20 вариантов выбора двух героев. И есть 5

вариантов, когда все три варда попадут к одному из героев. Таким образом, возможно  $10 + 20 + 5 = 35$  способов распределения 3 sentry вардов среди 5 героев. А общее количество способов распределения всех вардов между героями команды в соответствии с правилом произведения составит  $15 \cdot 35 = 525$ .

### **Игра «Карлсон, который живет на крыше»**

В мини-игре «Весы» из обучающей компьютерной игры «Карлсон, который живет на крыше» игроку на каждом уровне игры нужно подвесить все имеющиеся грузы так, чтобы весы пришли в равновесие (рис. 5–6). На уровне  $h$  весы представляют собой полное двоичное дерево высоты  $h$ , в котором число листьев-крючков равно  $2^h$ .

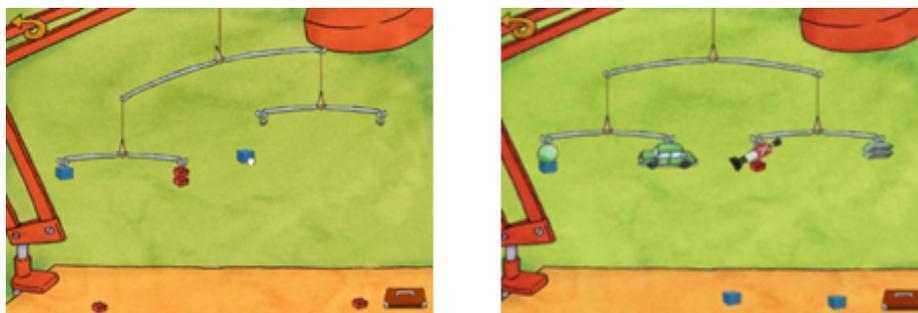


Рис. 5. Мини-игра «Весы», уровень 2

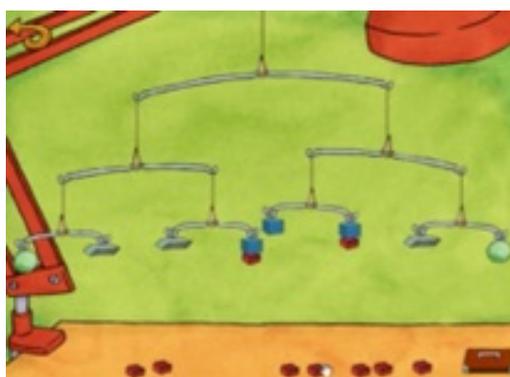


Рис. 6. Мини-игра «Весы», уровень 3

При каждой попытке игроку выдается новый набор грузов. Вес грузов игроку неизвестен, соотношение между весами грузов определяется

опытным путем. Минимальный вес имеет красная деталь конструктора, примем ее вес за 1. Тогда вес синего кубика равен 2, ластика — 3, зеленого шара — 4, солдатака — 5, машинки — 6.

На первом уровне весы имеют всего 2 крючка, и сначала набор грузов генерируется так, чтобы игрок мог соотнести веса грузов между собой, например ластик и три детали конструктора, шар и два кубика и т. п. Затем игра усложняется. Пусть вес грузов на каждом крючке равен 4. Как разработчик игры может сгенерировать общий набор грузов, который позволит уравновесить весы? Сколькими способами можно сгенерировать такие наборы?

Общий набор грузов для весов есть сумма наборов для каждого крючка. Если вес на каждом крючке должен быть равен 4, то количество различных наборов грузов для одного крючка равно **количеству разбиений числа 4 на слагаемые**. Таких разбиений 5. Далее нужно выбрать из 5 возможных наборов грузов 2. Если мы хотим, чтобы наборы грузов для крючков были обязательно различны (так интереснее), то число способов составить общий набор грузов равно числу сочетаний без повторений  $C_5^2 = 10$ .

На втором уровне с тем же весом для крючков ситуация будет аналогичной, число способов составить общий набор грузов будет равно  $C_5^4 = 5$ . Но при весе 4 для крючков на третьем уровне окажется, что различных наборов грузов у нас 5, а крючков 8. То есть различных наборов грузов для каждого крючка не хватает. Тогда мы допускаем возможность повторения наборов грузов на крючках, и количество способов составить общий набор грузов будет равно **числу сочетаний с повторениями**  $\bar{C}_5^8 = 495$ .

Генерация набора грузов для крючков сводится к генерации разбиений числа, равного суммарному весу грузов на крючке, на слагаемые. А количество способов сформировать общий набор грузов равно числу сочетаний с повторениями или без из числа разбиений по количеству крючков.

Разбиения числа на слагаемые можно сгенерировать, например, при помощи алгоритма генерации разбиений в обратном лексикографическом порядке [5, с. 446]. Число разбиений можно определить по рекуррентной формуле Эйлера [5, с. 451].

$$p(n) = p(n-1) + p(n-2) - p(n-5) - p(n-7) + \dots + (-1)^{q+1}(p(n - (3q^2 - q)/2) + p(n - (3q^2 + q)/2)) \dots$$

По этой формуле:

$$\begin{aligned} p(1) &= p(0) = 1, & p(2) &= p(1) + p(0) = 2, & p(3) &= p(2) + p(1) = 3, \\ p(4) &= p(3) + p(2) = 5, & p(5) &= p(4) + p(3) - p(0) = 7, \\ p(6) &= p(5) + p(4) - p(1) = 11. \end{aligned}$$

Пусть вес грузов для каждого крючка будет равен 6 (это вес самого тяжелого груза), тогда количество различных наборов грузов для крючков равно  $p(6) = 11$ . Если мы допускаем возможность повторения наборов грузов для крючков, то число способов составить общий набор грузов на первом уровне равно  $\bar{C}_{11}^2 = 66$ . Для второго уровня —  $\bar{C}_{11}^4 = 1001$ . Для третьего уровня —  $\bar{C}_{11}^8 = 43\,758$ . Очень много.

### Игра «Hollow Knight»

Задача составлена студентом группы 1216-ПИо Игнатом Туркиным и дополнена авторами. В приключенческой компьютерной игре «Hollow Knight» герой постепенно накапливает амулеты и ячейки, в которых амулеты можно разместить (рис. 7). Стартовое количество ячеек — 3, максимальное — 11. Амулеты могут быть четырех типов: А — занимают 1 ячейку, Б — 2 ячейки, В — 3 ячейки, Д — 4 ячейки.

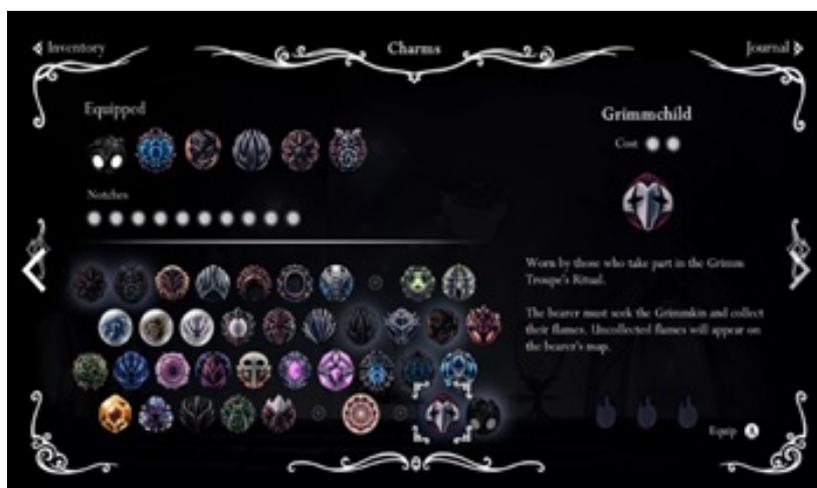


Рис. 7. Игра «Hollow Knight»

Предположим, на некотором начальном уровне игры у героя имеется 3 ячейки и 25 амулетов: 10 амулетов А, 9 амулетов Б и 6 амулет-

тов В. Сколькими способами герой может собрать комплект амулетов при условии, что все ячейки должны быть заполнены?

Порядок размещения амулетов в ячейках не имеет значения, поэтому варианты выбора амулетов: 1В или 1Б + 1А или 3А. Используя правила суммы и произведения и формулу числа сочетаний без повторов, определяем количество способов:  $6 + 9 \cdot 10 + C_{10}^3$ .

При максимальном развитии у главного героя имеется 11 ячеек под амулеты и 39 всевозможных амулетов: амулетов А — 10, амулетов Б — 16, амулетов В — 9, амулетов Д — 4. Сколькими способами герой может собрать комплект амулетов при условии, что все ячейки должны быть заполнены?

Количество вариантов выбора амулетов равно **числу разбиений числа 11 на слагаемые, не превышающие 4**. Для каждого из 26 разбиений вычисляем количество способов комплектации амулетов, затем складываем их. Например, для разбиения  $4 + 4 + 3$  ( $2Д + 1В$ ) количество способов равно  $C_4^2 \cdot C_9^1$ . Конечно, в этом случае ручной счет утомительный, и желательно составить программу генерации разбиений числа на слагаемые. Всего получается 1 189 411 различных способов собрать комплект амулетов.

### 3. Обсуждение

Задание «Сконструировать задачу по комбинаторике на основе...» относится к наивысшей категории когнитивных процессов модифицированной таксономии Блума — помнить, понимать, применять, анализировать, критиковать, **создавать** [6]. Некоторым студентам сложно выполнить такое задание. Им можно предложить составить задачу, аналогичную некоторому примеру. В этом случае задание будет соответствовать категории **применять**, а при некотором развитии примера категории **анализировать**, что тоже очень хорошо.

Рассмотрим два модельных примера, на основе которых студенты могут составить свои собственные задачи.

#### Игры «Найди различия»

В таких играх нужно найти отличия между двумя практически идентичными картинками (рис. 8).

Предположим, мы хотим разработать игру для детей или пожилых людей, в которой на основе каждой имеющейся картинке будут автоматически генерироваться несколько отличающихся деталями картинок (рис. 9). То есть у игрока при желании будет несколько попыток

на одном уровне с одной и той же картинкой. На первом уровне между картинками должно быть только одно отличие, на втором уровне — два. Менять можно цвет 6 желтых точек, цвет 10 внутренних участков на крыльях (сиреневый, оранжевый и зеленый на верхних крыльях, красный и зеленый на нижних). Сколько разных картинок можно сгенерировать для первого и второго уровней?



Рис. 8. Яндекс игры: Найди отличия: Головоломка



Рис. 9. Картинка-основа (источник – Яндекс-картинки)

На первом уровне можно сгенерировать по правилу сложения  $6 + 10 = 16$  разных картинок. На втором уровне по правилу произведения можно сгенерировать  $16 \cdot 15 = 240$  картинок. Если наложить дополнительное условие, что меняться должны объекты разных типов, т. е. изменяется 1 точка и 1 область на крыле, то получим  $6 \cdot 10 = 60$  возможных вариантов генерации картинок. Вариантов много, поэтому

можно добавить еще условия: например, один изменяемый объект должен быть справа, а второй — слева. В этом случае можно сгенерировать  $6 \cdot 5 = 30$  разных картинок.

На основе этого примера студенты могут составить свои задачи, изменив картинку, определив набор изменяющихся элементов и дополнительные условия. Задача может усложниться за счет добавления третьего уровня, а при желании и последующих.

### Игры с лабиринтами

При создании компьютерных игр широко применяется процедурная генерация игровых миров и их деталей. Существует большое количество алгоритмов генерации лабиринтов. Создаваемые при помощи различных алгоритмов лабиринты отличаются количеством проходов, длиной коридоров и тупиков, текстурой и т. п. Все алгоритмы можно разделить на те, в которых стены удаляются, и те, в которых стены строятся.

Рассмотрим самый простой алгоритм генерации лабиринтов — алгоритм двоичного дерева. Имеется прямоугольная сетка, в которой границы клеток являются стенами. Выбираем два направления движения, пусть это будут «вправо» и «вверх». В каждой клетке случайным образом выбираем одно из направлений и стираем соответствующую стену. В крайних справа и крайних сверху клетках сетки выбор направления определен единственным образом. В результате получается идеальный лабиринт (лабиринт с одиночным соединением), т. е. лабиринт с единственным проходом без циклов и изолированных участков. Сколько различных лабиринтов может быть сгенерировано при помощи алгоритма двоичного дерева на сетке размера  $n$  на  $n$ ?

Поскольку такие лабиринты всегда имеют два коридора вдоль верхней и вдоль правой границ, то количество вариантов можно определить, как число размещений с повторениями —  $2n - 1$ .

На рис. 10 представлен лабиринт, построенный при помощи алгоритма двоичного дерева на сетке 5 на 5. В нем имеется пять тупиков. Предположим, что в конце каждого тупика может быть размещен клад. Сколько существует вариантов распределения кладов по тупикам, если хотя бы один клад в лабиринте должен быть обязательно? Все клады одинаковые.

Всего вариантов распределения кладов — число размещений с повторениями —  $2^5$ . Единственный вариант распределения кладов, который нас не устраивает — случай, когда клада нет ни в одном тупике. Следо-

вательно, имеется  $2^5 - 1 = 31$  способ распределить клады в лабиринте требуемым образом.

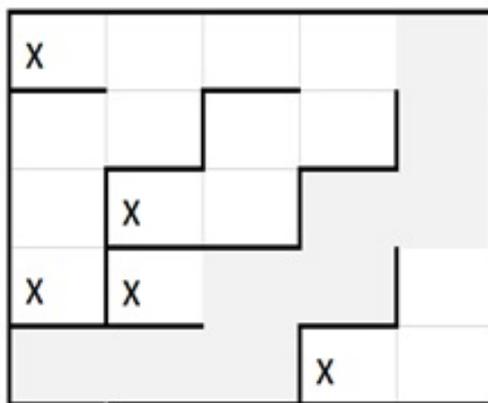


Рис. 10. Лабиринт, построенный при помощи алгоритма двоичного дерева

В этом примере можно поменять приблизительно все: алгоритм генерации лабиринта (а значит, изменится структура лабиринта), размер сетки, клады могут быть разных типов, количество кладов может меняться.

### **Заключение**

В 2022–2023 годах Н. Н.Бабиковой было проведено анкетирование студентов направления подготовки «Прикладная информатика». Анкета содержала в том числе вопросы об эффективности тех или иных видов учебной деятельности в процессе обучения математическим дисциплинам. В 2022 году на вопрос с открытым ответом — «Перечислите виды учебной деятельности на аудиторных занятиях, которые помогают вам лучше запоминать / понимать новую информацию» — самым распространенным ответом был «Анализ / примеры применения в реальной деятельности», его дали 43 % студентов [7, с. 39]. В 2023 году этот вопрос был задан в закрытой форме, 52 % студентов выбрали ответ — «Решение задач с реальными данными». А что может быть для современных студентов более реальным, чем виртуальный мир компьютерных игр?

Студенты направления подготовки «Прикладная информатика» положительно отнеслись как к решению комбинаторных задач по мотивам компьютерных игр на практических и лабораторных занятиях, так и к

заданию на самостоятельную разработку задач. В 2023 году студент группы 1216-ПИо Николай Мельник составил целый комплект таких задач в рамках самостоятельной научно-исследовательской работы и подготовил доклад для выступления на научной студенческой конференции.

Планируется использование составленных авторами задач для студентов направлений «Прикладная математика и информатика», «Математика и компьютерные науки» (дисциплины «Дискретная математика», «Комбинаторные алгоритмы», «Приложения дискретной математики в компьютерных науках»).

Для студентов информационных специальностей разработка компьютерных игр является одним из возможных видов будущей профессиональной деятельности. Математические методы и модели широко применяются в геймдизайне [8; 9], поэтому в дальнейшем авторы считают целесообразным разработку задач на тему компьютерных игр, сформулированных с точки зрения геймдизайнера: таких, как приведенная выше задача про весы в игре «Карлсон, который живет на крыше».

### Благодарности

Авторы выражают благодарность студентам направления «Прикладная информатика» СГУ им. Питирима Сорокина за участие в исследовании.

### Список источников

1. Тюменева Ю. А., Вальдман А. И., Карной М. Что дают предметные знания для умения применять их в новом контексте. Первые результаты сравнительного анализа TIMSS-2011 и PISA-2012 // *Вопросы образования*. 2014. № 1. С. 8–24. DOI 10.17323/1814-9545-2014-1-8-24.
2. Тюменева Ю. А., Шкляева И. В. Два подхода к пониманию «применения знаний»: трансфер и моделирование. Обзор литературы и критика // *Вопросы образования*. 2016. № 3. С. 8–33. DOI 10.17323/1814-9545-2016-3-8-33.
3. Tyumeneva Y. A., Goncharova M. V. Following the Template: Transferring Modeling Skills to Nonstandard Problems // *Russian*

*Education & Society*. 2017. Vol. 59. No 5–6. Pp. 298–318. DOI 10.1080/10609393.2017.1408370.

4. Виленкин Н. Я., Виленкин А. Н., Виленкин П. А. Комбинаторика. М.: ФИМА, МЦНМО, 2006. 400 с.
5. Кнут Д. Искусство программирования. Том 4, А. Комбинаторные алгоритмы. Часть 1 : пер. с англ. М.: ООО «И.Д. Вильямс», 2013. 960 с.
6. Бабикова Н. Н. Согласование методов оценивания и результатов обучения на основе таксономии // *Образование и педагогические науки в XXI веке: актуальные вопросы, достижения и инновации : сборник статей II Международной научно-практической конференции : в 2 частях*. Пенза: Наука и Просвещение (ИП Гуляев Г. Ю.), 2017. Ч. 1. С. 93–100.
7. Бабикова Н. Н. Обучение в цифровую эпоху: помнить или гуглить // *Вестник Сыктывкарского университета. Серия 1: Математика. Механика. Информатика*. 2022. Вып. 3 (44). С. 33–46. DOI 10.34130/1992-2752\_2022\_3\_33.
8. Патрашов А. С. Математическое руководство по созданию компьютерных игр: справочник. Екатеринбург: Интернет-издательство Ridero, 2016. 316 с.
9. Патрашов А. С. Применение высшей математики в дизайне компьютерных игр // *Системный администратор*. 2020. № 5 (210). С. 46–49.

## References

1. Tyumeneva Yu. A., Val'dman A. I., Karnoi M. What Does Subject Knowledge Give for Its Applying in New Context. The First Results from Studies TIMSS-2011 and PISA-2012. *Voprosy obrazovaniya* [Questions of education]. 2014. No 1. Pp. 8–24. DOI 10.17323/1814-9545-2014-1-8-24. (In Russ.)
2. Tyumeneva Yu. A., Shklyieva I. V. Two Approaches to the Concept of Knowledge Application: Transfer and Modeling. Overview

- and Criticism. *Voprosy obrazovaniya* [Questions of education]. 2016. No 3. Pp. 8–33. DOI 10.17323/1814-9545-2016-3-8-33. (In Russ.)
3. **Tyumeneva Y. A., Goncharova M. V.** Following the Template: Transferring Modeling Skills to Nonstandard Problems. *Russian Education & Society*. 2017. No 59 (5–6). Pp. 298–318. DOI 10.1080/10609393.2017.1408370.
  4. **Vilenkin N. Ya., Vilenkin A. N., Vilenkin P. A.** *Kombinatorika* [Combinatorics]. Moscow: FIMA, MCzNMO, 2006. 400 p. (In Russ.)
  5. **Knuth D.** *Iskusstvo programmirovaniya. Tom 4, A. Kombinatornyye algoritmy* [The Art of Computer Programming, Vol. 4, A: Combinatorial Algorithms]. Moscow: I.D. Williams LLC, 2013. Part 1. 960 p. (In Russ.)
  6. **Babikova N. N.** Aligning assessment methods with learning outcomes based on taxonomy. *Obrazovaniye i pedagogicheskiye nauki v XXI veke: aktual'nyye voprosy, dostizheniya i innovatsii : sbornik statey II Mezhdunarodnoy nauchno-prakticheskoy konferentsii : v 2 chastyah* [Education and pedagogical science in the XXI century: current issues, achievements and innovations : collection of articles of the international scientific and practical conference : in 2 parts]. Penza, 2017. Part. 1. Pp. 93–100. (In Russ.)
  7. **Babikova N. N.** Education in the digital age: remember or google *Vestnik Syktyvskarskogo universiteta. Seriya 1: Matematika. Mekhanika. Informatika* [Bulletin of Syktyvkar University. Series 1: Mathematics. Mechanics. Computer science]. 2022. No 3 (44). Pp. 33–46. DOI 10.34130/1992-2752\_2022\_3\_33.(In Russ.)
  8. **Patrashov A. S.** *Matematicheskoe rukovodstvo po sozdaniyu komp'yuterny'x igr: spravochnik* [A mathematical guide to creating computer games: guide.] Ekaterinburg: Internet-izdatel'stvo Ridero, 2016. 316 p. (In Russ.)
  9. **Patrashov A. S.** Application of higher mathematics in computer game design. *Sistemnyy administrator* [System Administrator]. 2020. No 5 (210). Pp. 46–49. (In Russ.)

Сведения об авторах / Information about authors

Бабикова Надежда Николаевна / Nadezhda N. Babikova

к.пед.н., доцент, доцент кафедры прикладной информатики /  
PhD (Pedagogy), Associate Professor, Associate Professor of Applied  
Informatics Department

Сыктывкарский государственный университет имени Питирима Сорокина / Pitirim Sorokin Syktyvkar State University

167001, Россия, г. Сыктывкар, Октябрьский пр., 55 / 167001, Russia,  
Syktyvkar, Oktyabrsky Ave., 55

Котелина Надежда Олеговна / Nadezhda O. Kotelina

к.ф.-м.н., доцент кафедры прикладной математики и компьютерных наук / Ph.D. in Physics and Mathematics, Associate Professor of Department of Applied Mathematics and Computer Science

Сыктывкарский государственный университет имени Питирима Сорокина / Pitirim Sorokin Syktyvkar State University

167001, Россия, г. Сыктывкар, Октябрьский пр., 55 / 167001, Russia,  
Syktyvkar, Oktyabrsky Ave., 55

Валуева Мария Александровна / Marija A. Valueva

обучающийся бакалавриата/ undergraduate student

Санкт-Петербургский государственный институт культуры / St. Petersburg State University of Culture

191186, г. Санкт-Петербург, Дворцовая наб., д. 2 / 191186, St. Petersburg,  
Dvortsovaia embankment, 2

Старцев Николай Андреевич / Nikolaj A. Startsev

обучающийся бакалавриата/ undergraduate student

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» имени В. И. Ульянова (Ленина) / St. Petersburg State Electrotechnical University "LETI" named after V. I. Ulyanov (Lenin)

197022, Россия, г. Санкт-Петербург, ул. Профессора Попова, 5, литера Ф / 197022, Russia, St. Petersburg, Professora Popova str., 5, litera F

Статья поступила в редакцию / The article was submitted 16.02.2024

Одобрено после рецензирования / Approved after reviewing 26.02.2024

Принято к публикации / Accepted for publication 11.03.2024