

DOI 10.26105/SSPU.2022.76.1.012

УДК 373.3.016:004

ББК 74.263.2с5

С.Ф. ГРОМОВА

**ОСОБЕННОСТИ РЕШЕНИЯ ЗАДАЧ НА
ИГРОВЫЕ СТРАТЕГИИ СРЕДСТВАМИ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
В ШКОЛЬНОМ КУРСЕ ИНФОРМАТИКИ**

S.F. GROMOVA

**FEATURES OF SOLVING PROBLEMS
OF GAME STRATEGIES BY MEANS
OF INFORMATION TECHNOLOGIES
IN THE SCHOOL COURSE
OF COMPUTER STUDIES**

В статье описываются способы решения задач на игровые стратегии с использованием информационных технологий. Учебная тема «Поиск выигрышных стратегий» рассматривается исключительно в курсе углубленного изучения информатики в 11 классе. Задачи на игровые стратегии традиционно включаются в состав контрольно-измерительных материалов единого государственного экзамена по информатике. В настоящее время в школьных учебниках различных авторов предложен только аналитический способ решения подобных задач. Введение в 2020 году Единого государственного экзамена по информатике в компьютерной форме, поставил перед школьными учителями и методистами в области информатики проблему поиска оптимальных вариантов решения с использованием прикладного программного обеспечения или инструментария программирования, в зависимости от уровня подготовки обучающихся. В качестве одного из вариантов решения данной проблемы, автор предлагает два способа решения задач на игровые стратегии на примере задачи с камнями. Первый вариант — решение задачи средствами табличного процессора предназначен для обучающихся, не имеющих опыта программирования, второй вариант — решение задачи с использованием языка программирования Python. Данные способы решения задач на игровые стратегии были успешно апробированы в процессе подготовки выпускников образовательных организаций г. Сургута Ханты-Мансийского автономного округа-Югры к Единому государственному экзамену по информатике в компьютерной форме. Результаты, полученные по итогам экзамена, показали эффективность предложенных способов решения, позволяющих визуализировать информацию и оптимизировать обработку больших массивов данных средствами информационных технологий. Предложенные способы решения будут полезны будущим учителям математики и информатики (уровень бакалавриата).

This article describes ways to solve problems of game strategies using information technology. The educational topic «Search for winning strategies» is considered only in the course of advanced study of computer studies in grade 11. Tasks for game strategies are traditionally included in the control and measuring materials of the unified state exam in computer studies. At present, only an analytical method for solving such problems is proposed in school textbooks by various authors. The introduction of the unified state exam in computer science in 2020 on computers posed the problem for school teachers and methodologists in the field of computer studies to find the best options for solving such problems using application software or programming tools, depending on the training level of students. As one of the options for solving this problem, the

author offers two ways to solve problems of game strategies using the example of a problem with stones, the first option is intended for students who do not have programming experience and includes solving the problem using a spreadsheet processor, the second option is solving the problem using the programming language Python. These methodological recommendations for solving problems of game strategies were used in the preparation of graduates for the unified state exam in computer studies on computers for students in schools in the city of Surgut, Khanty-Mansiisk Autonomous district-Yugra. The results obtained on the basis of the exam showed the effectiveness of the proposed solutions, which allow visualizing information and optimizing the processing of great numbers of data by means of information technology.

КЛЮЧЕВЫЕ СЛОВА: методика преподавания информатики; игровые стратегии; решение задач по информатике; единый государственный экзамен.

KEY WORDS: technique of teaching computer studies; game strategies; solving problems in computer studies; unified state exam

ВВЕДЕНИЕ. Школьная информатика является самой молодой и наиболее динамично развивающейся школьной дисциплиной, что обусловлено высокими темпами развития информационных технологий в современном мире. Это касается не только изменения содержания самого предмета, но и методов решения задач. Тема «Поиск выигрышных стратегий», пришедшая из математики в углубленный курс изучения информатики, базируется на достаточно высоком уровне математического развития и предполагает формирование у обучающихся понимания сущности игровой стратегии (понятий выигрышной и проигрышной позиций), умения анализировать ситуацию и строить дерево игры, прогнозировать результаты собственных действий и возможные реакции противника. Теоретические основы теории игр и игровых стратегий, на которых построены алгоритмы решения задач по информатике, представлены в работах Дж. Фон Неймана, Дмитриевой Ю.Ю., Григорьевой Н.Д., Петросьяна Л.А., Зенкевич Н.А., Губко М.В., Костромина А.В.

Задачи данного типа относятся к заданиям повышенного уровня сложности и традиционно включаются в содержание Единого государственного экзамена по информатике и информационно-коммуникационным технологиям (далее — ЕГЭ). В 2021 году выпускники российских школ, принявшие участие в едином государственном экзамене по информатике и ИКТ в компьютерной форме (далее — КЕГЭ), показали, что 71,7% (в ХМАО-Югре — 68,6%) умеют анализировать алгоритм логической игры; 62,9% (в ХМАО-Югре — 58,8%) умеют найти выигрышную стратегию; 46,6% в (ХМАО-Югре — 42,7%) умеют построить дерево игры по заданному алгоритму и найти выигрышную стратегию. Результаты, продемонстрированные выпускниками нашего округа, в среднем, на 4% ниже российских показателей. Анализ решаемости заданий 19–21 позволяет сделать вывод о том, что обучающиеся не рассматривают триаду задач как единую систему логически связанных между собой компонентов, поэтому достаточно успешно справляются с локальными компонентами — заданиями, не ориентированными на обработку значительного массива данных и испытывают серьезные затруднения при решении задач, требующих большей глубины анализа и обработки большого массива числовых значений (решаемость снижается в среднем на 20%). Теоретические основы теории игр, поиск выигрышных стратегий, аналитический способ решения в сжатой форме представлены в учебно-методических комплексах по информатике и ИКТ для 11 класса Хеннера Е.К., Семакина И.Г., Шеиной Т.Ю., Угриновича Н.Д., Полякова К.Ю., Еремина Е.А.

Решение задач по поиску выигрышных стратегий, изложенное в школьных учебниках, требует от учеников производить «вручную» рекурсивные переборы, рассматривать различные варианты ходов в рамках учебного предмета «Информатика», в котором не только основным средством обучения, но и объектом изучения является компьютер.

В настоящее время учителя информатики А. Сидоров, А. Кабанов, Д. Муфаззалов, В. Бабий и др. предлагают различные варианты решения задач на поиск выигрышных стратегий с использованием средств информационных технологий [1].

ЦЕЛЬЮ данного исследования является разработка способов решения задач на игровые стратегии с использованием средств информационных технологий.

Новизна и практическая значимость представленного исследования заключаются в том, что решение задач по теории игр с использованием табличных процессоров и (или) языков программирования позволит обучающимся автоматизировать процесс обработки больших массивов данных и оптимизировать процесс решения в целом. В ходе практического применения предложенных подходов к решению задач школьники смогут смоделировать процесс игры, визуализировать математическую модель, учесть необходимые условия, провести компьютерный эксперимент. Предложенные способы решения задач на поиск игровых стратегий могут быть использованы учителями информатики общеобразовательных организаций при подготовке обучающихся к решению заданий 19-21 КЕГЭ по информатике.

МАТЕРИАЛЫ И МЕТОДЫ ИССЛЕДОВАНИЯ. В спецификации контрольно-измерительных материалов ЕГЭ по информатике (далее — КИМ) в 2019 году на выполнение задания 26 (что соответствует заданиям 19-21 КЕГЭ) отводилось 30 минут, а в 2022 году время сократилось всего на 5 минут, с учетом того, что в предыдущие годы это задание относилось ко второй части ЕГЭ и требовало развернутого описания соответствующих стратегий игроков, в то время как в компьютерном варианте экзамена необходимо внести только правильные ответы, кроме того, для решения задачи (имеем в виду триаду 19-21) не предполагается использование информационных технологий, что, на наш взгляд, противоречит целям компьютерного ЕГЭ по информатике, предполагающим проверку уровня освоения выпускниками школ информационно-технологических компетенций [10,14].

Следует отметить, что в основе представленного решения лежит классический алгоритм, суть которого сводится к переводу противника в проигрышную позицию. Решение подобных задач должно начинаться с четкого усвоения обучающимися основных теоретических понятий и аналитического решения задачи на одну кучу камней, при которой количество рекурсивных переборов не велико [4,5]. Только на основе хорошо усвоенной обучающимися математической модели можно оптимизировать задачу средствами информационных технологий.

Рассмотрим два способа решения задачи №31, наиболее распространенного типа задач «на камни» КЕГЭ по информатике, опубликованную на официальном сайте К.Ю. Полякова [1].

Первый способ — решение задачи с использованием возможностей табличного процессора Excel.

1.1. Анализ условия задачи.

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежат две кучи камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в одну из куч один камень или увеличить количество камней в куче в три раза. Например, пусть в одной куче 5 камней, а в другой 9 камней; такую позицию мы будем обозначать (5, 9). За один ход из позиции (5, 9) можно получить любую из четырех позиций: (6, 9), (15, 9), (5, 10), (5, 27).

Чтобы делать ходы, у каждого игрока есть неограниченное количество камней. Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 79. Победителем считается игрок, сделавший последний ход, то есть первым получивший позицию, в которой в кучах будет 79 или больше камней.

В начальный момент в первой куче было 6 камней, во второй куче — S камней, $1 \leq S \leq 72$.

Будем считать, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника.

Из условия задачи становится ясно, что первоначально в двух кучах было $(6, S)$ камней (первая цифра означает количество камней в первой куче, вторая — количество камней во второй куче).

Условием победы является суммарное количество камней в двух кучах ≥ 79 .

Согласно теории игр, для выигрывающего игрока будем учитывать только выигрышные позиции, а для проигрывающего игрока будем учитывать все возможные позиции.

1.2. Визуализация условия задачи. Отметим на игровом поле:

- возможные ходы игроков;
- сумму камней двух куч, которая определяет выигрыш игрока;
- номер выполняемого задания.

Определим названия столбцов (названия столбцов могут отличаться от представленных, также как и их расположение). Обозначим количество камней в первой куче (из условия задачи) и второй (первоначально поставим любое число, т.к. оно будет изменяться в процессе решения задачи) куче (рис. 1).

A6						
	A	B	C	D	E	F
1	Возможные ходы:	+1				
2		*3				
3	Выигрыш:	79				
4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	10			
7						
8						
9						
10						

Рис. 1. Начальные условия игры на игровом поле

1.3. Ввод формул в соответствии с возможными ходами игроков.

После определения позиций на поле игры, необходимо ввести формулы, по которым будет вычисляться количество камней. В соответствии с условием задачи первый игрок может получить четыре варианта позиций (увеличение камней на 1 и в 6 раз, соответственно, в первой и второй кучах): $(6+1; S)$; $(6; S+1)$; $(6*3; S)$; $(6; 3S)$ (рис. 2).

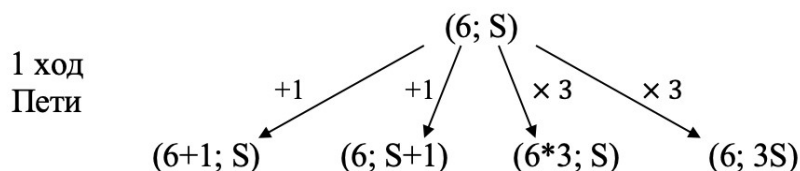


Рис. 2. Возможные позиции после первого хода Пети

Используя мастер функций табличного процессора, введем формулы определения количества камней в первой и второй кучах соответственно для столбцов D6: D9 и E6: E9, отражающих положение — «1 ход Пети», представленные в таблице 1.

Таблица 1 Последовательность заполнения столбца «1 ход Пети»

1							2						
СТАНДО... fx =B6+1							СТАНДО... fx =C6						
A	B	C	D	E	F		A	B	C	D	E	F	
1	Возможные ходы:	+1					1	Возможные ходы:	+1				
2		*3					2		*3				
3	Выигрыш:	79					3	Выигрыш:	79				
4							4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани		5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	10	=B6+1			6	19.	6	10	7=C6		
7							7						
8							8						
9							9						
10							10						

3							4						
СТАНДО... fx =B6+1							СТАНДО... fx =C6						
A	B	C	D	E	F		A	B	C	D	E	F	
1	Возможные ходы:	+1					1	Возможные ходы:	+1				
2		*3					2		*3				
3	Выигрыш:	79					3	Выигрыш:	79				
4							4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани		5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	10	=B6+1			6	19.	6	10	7=C6		
7							7						
8							8						
9							9						
10							10						

5							6						
СТАНДО... fx =B6+1							СТАНДО... fx =C6						
A	B	C	D	E	F		A	B	C	D	E	F	
1	Возможные ходы:	+1					1	Возможные ходы:	+1				
2		*3					2		*3				
3	Выигрыш:	79					3	Выигрыш:	79				
4							4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани		5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	10	=B6+1			6	19.	6	10	7=C6		
7							7						
8							8						
9							9						
10							10						

7							8						
СТАНДО... fx =B6+1							СТАНДО... fx =C6						
A	B	C	D	E	F		A	B	C	D	E	F	
1	Возможные ходы:	+1					1	Возможные ходы:	+1				
2		*3					2		*3				
3	Выигрыш:	79					3	Выигрыш:	79				
4							4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани		5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	10	=B6+1			6	19.	6	10	7=C6		
7							7						
8							8						
9							9						
10							10						

Заполнение столбца «1 ход Вани» происходит с использованием формулы «=макс(D6: E6)*3+мин(D6: E6)», что означает сумму первой и второй куч с увеличением большей из них в три раза, так как согласно игровой стратегии, Ваня использует ход с максимально возможным увеличением количества камней для обеспечения гарантированного выигрыша (рис. 3).

СТАНДО...		✖	✓	f_x	$=\text{макс}(\text{D6:E6}) * 3 + \text{мин}(\text{D6:E6})$			
	A	B	C	D	E	F	G	H
1	Возможные ходы:		+1					
2			*3					
3	Выигрыш:		79					
4								
5		1 куча	2 куча	1 ход Пети	1 ход Вани			
6	19.	6	10	7	10	$=\text{макс}(\text{D6:E6}) * 3 + \text{мин}(\text{D6:E6})$		
7				6	11			
8				18	10			
9				6	30			
10								

Рис. 3. Обеспечение выигрышной стратегии Вани

Использование инструмента «Условное форматирование» позволяет визуализировать выигрыш, выделив ячейку соответствующим цветом, в нашем случае — зеленый цвет означает выигрышную позицию. В результате выполнения предложенных действий получаем расчетную модель, с помощью которой решаются задания 19–21 КЕГЭ (рис. 4).

С6		✖	✓	f_x	10			
	A	B	C	D	E	F		
1	Возможные ходы:		+1					
2			*3					
3	Выигрыш:		79					
4								
5		1 куча	2 куча	1 ход Пети	1 ход Вани			
6	19.	6	10	7	10	37		
7				6	11	39		
8				18	10	64		
9				6	30	96		
10								

Рис. 4. Использование условного форматирования для визуализации выигрышной позиции

1.4. Решение задачи № 19.

Известно, что Ваня выиграл своим первым ходом после неудачного первого хода Пети. Укажите минимальное значение S, когда такая ситуация возможна.

Аналитическое решение:

Минимальное значение S, при котором Ваня может выиграть после неудачного хода Пети, может быть обеспечено только утроением количества камней либо в первой, либо во второй кучах.

Рассмотрим два варианта: утроение количества камней в первой и во второй кучах соответственно и выберем минимальное значение:

Первый вариант: неудачный ход Пети (первого игрока) заключается в том, что он увеличивает количество камней во второй куче в три раза, далее такое же действие производит второй игрок. В итоге суммарное количество камней должно соответствовать условию выигрыша. Из полученного неравенства находим количество камней во второй куче.

Ход игры: $(6; S) \rightarrow (6; 3S) \rightarrow (6; 9S) \rightarrow 6 + 9S \geq 79 \rightarrow 9S \geq 73 \rightarrow S \geq 9$. Дерево игры представлено на рис. 5.

Второй вариант: повторяем тот же алгоритм для первой кучи.

Ход игры: $(6; S) \rightarrow (18; S) \rightarrow (54; S) \rightarrow 54 + S \geq 79 \rightarrow S \geq 25$

Дерево игры представлено на рис. 6.

По условию необходимо указать минимальное значение S, следовательно $S = 9$.

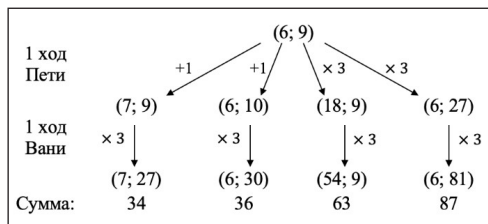


Рис. 5. Утроение камней во второй куче

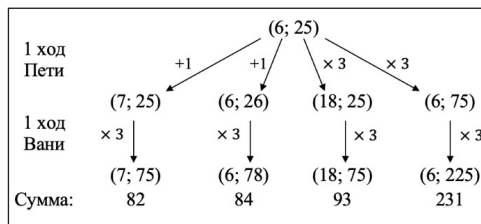


Рис. 6. Утроение камней в первой куче

Компьютерное решение представлено на рис. 7 и рис. 8.

С6	A	B	C	D	E	F
1	Возможные ходы:	+1				
2		*3				
3	Выигрыш:	79				
4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	9	7	9	34
7				6	10	36
8				18	9	63
9				6	27	87
10						

Рис. 7. Компьютерная модель

С6	A	B	C	D	E	F
1	Возможные ходы:	+1				
2		*3				
3	Выигрыш:	79				
4						
5		1 куча	2 куча	1 ход Пети	1 ход Вани	
6	19.	6	25	7	25	82
7				6	26	84
8				18	25	93
9				6	75	231
10						

Рис. 8. Компьютерная модель

1.5. Решение задачи № 20. Условие задачи

Для игры, описанной в задании № 19, найдите все такие значения S , при которых у Пети есть выигрышная стратегия, причём Петя не может выиграть первым ходом, но может выиграть своим вторым ходом независимо от того, как будет ходить Ваня. Найденные значения запишите в ответе в порядке возрастания.

Для решения задачи № 20 необходимо воспользоваться ранее составленной таблицей для задачи № 19 с внесением некоторых корректив. Так как Петя, по условию задачи, не может гарантированно выиграть первым ходом, но выигрывает при любом ходе Вани, который в данном случае является проигравшим игроком, то для него надо рассмотреть (проанализировать) все ходы.

Продублируем ранее скопированные значения таблицы из задания 19 ниже ещё три раза, так как первый ход Пети может определяться следующими исходами: $(6+1; S)$; $(6; S+1)$; $(6*3; S)$; $(6; 3S)$, соответственно исправим их для столбцов D и E (см. рис. 9).

СТАНДО...	A	B	C	D	E	F	G	H
11								
12	20.	1 куча	2 куча	1 ход Пети	1 ход Вани	2 ход Пети		
13		6	10	7	10	8	10	38
14						7	11	40
15						21	10	73
16						7	30	97
17				6	11	7	11	40
18						6	12	42
19						18	11	65
20						6	33	105
21				18	10	19	10	67
22						18	11	65
23						54	10	172
24						18	30	108
25				6	=C13*3	7	0	21
26						6	1	19
27						18	0	54
28						6	0	18
29								

Рис. 9. Компьютерная модель второго хода Пети

СТАНДО...	A	B	C	D	E	F	G	H	I
11									
12	20.	1 куча	2 куча	1 ход Пети	1 ход Вани	2 ход Пети	Проверка		
13		6	10	7	10	8	38	=F13+G13	
14						7	11	40	
15						21	10	73	
16						7	30	97	
17				6	11	7	11	40	
18						6	12	42	
19						18	11	65	
20						6	33	105	
21				18	10	19	10	67	
22						18	11	65	
23						54	10	172	
24						18	30	108	
25				6	30	7	30	97	
26						6	31	99	
27						18	30	108	
28						6	90	276	
29									

Рис. 10. Ввод формулы для проверки результатов

021	A	B	C	D	E	F	G	H	I
11									
12	20.	1 куча	2 куча	1 ход Пети	1 ход Вани	2 ход Пети	Проверка		
13		6	10	7	10	8	10	38	18
14						7	11	40	18
15						21	10	73	31
16						7	30	97	37
17				6	11	7	11	40	18
18						6	12	42	18
19						18	11	65	29
20						6	33	105	39
21				18	10	19	10	67	29
22						18	11	65	29
23						54	10	172	64
24						18	30	108	48
25				6	30	7	30	97	37
26						6	31	99	37
27						18	30	108	48
28						6	90	276	96
29									

Рис. 11. Результат проверки после второго хода Пети

C13	A	B	C	D	E	F	G	H	I
11									
12	20.	1 куча	2 куча	1 ход Пети	1 ход Вани	2 ход Пети	Проверка		
13		6	8	7	8	8	8	32	16
14						7	9	34	16
15						21	8	71	29
16						7	24	79	31
17				6	9	7	9	34	16
18						6	10	36	16
19						18	9	63	27
20						6	27	87	33
21				18	8	19	8	65	27
22						18	9	63	27
23						54	8	170	62
24						18	24	90	42
25				6	24	7	24	79	31
26						6	25	81	31
27						18	24	90	42
28						6	72	222	78
29									

Рис. 12. Выигрыш Пети вторым ходом

C13	A	B	C	D	E	F	G	H	I
11									
12	20.	1 куча	2 куча	1 ход Пети	1 ход Вани	2 ход Пети	Проверка		
13		6	20	7	20	8	20	68	28
14						7	21	70	28
15						21	20	83	41
16						7	60	187	67
17				6	21	7	21	70	28
18						6	22	72	28
19						18	21	81	39
20						6	63	195	69
21				18	20	19	20	79	39
22						18	21	81	39
23						54	20	182	74
24						18	60	198	78
25				6	60	7	60	187	67
26						6	61	189	67
27						18	60	198	78
28						6	180	546	186
29									

Рис. 13. Выигрыш Пети вторым ходом

В условии задачи №20 сказано, что Петя может выиграть своим вторым ходом независимо от игры Вани. При разработке компьютерной модели, следует обратить внимание обучающихся на то, что при подборе необходимого числа камней следует внимательно анализировать соответствующие позиции Вани (чтобы не пропустить его выигрыш первым ходом). Для этого добавим в таблицу столбец под названием «Проверка» и используем формулу «=F13+G13» (сумма камней первой и второй куч в ходе Вани), которая обезопасит от неправильного ответа.

Повторим третий шаг из этапа 3 и также воспользуемся автозаполнением для заполнения всех ячеек столбца и условным форматированием с использованием тех же параметров, что позволит отследить выигрыш Вани первым ходом, а также выделить его другим цветом (в нашем случае — красным). Результаты представлены ниже на рисунках 10 и 11.

Финальный шаг предполагает использование метода подбора числа путем изменения количества камней во второй куче.

После подстановки различных значений получаем следующие ответы:

выигрыш Пети вторым ходом при $S = 8$, где его первый ход (6; 24);

выигрыш Пети вторым ходом при $S = 20$, где его первый ход (18; 20);

выигрыш Пети вторым ходом при $S = 23$, где его первый ход (6; 24).

Таким образом, гарантированный выигрыш Пети вторым ходом обеспечивают три значения (см. рис. 12, 13, 14).

C13	A	B	C	D	E	F	G	H	I
11									
12	20.	1 куча	2 куча	1 ход Пети	1 ход Вани	2 ход Пети	Проверка		
13		6	23	7	23	8	23	77	31
14						7	24	79	31
15						21	23	90	44
16						7	69	214	76
17				6	24	7	24	79	31
18						6	25	81	31
19						18	24	90	42
20						6	72	222	78
21				18	23	19	23	88	42
22						18	24	90	42
23						54	23	185	77
24						18	69	225	87
25				6	69	7	69	214	76
26						6	70	216	76
27						18	69	225	87
28						6	207	627	213
29									

Рис. 14. Выигрыш Пети вторым ходом

1.6. Решение задачи № 21. Условие задачи.

Для игры, описанной в задании 19, укажите максимальное значение S, при котором у Вани есть выигрышная стратегия, позволяющая ему выиграть при любой игре Пети.

Для решения данной задачи достаточно таблицы из задачи 20, но с некоторыми дополнениями:

1) Необходимо скопировать таблицу из задачи 20. Так как в условии сказано, что у Вани есть выигрышная стратегия, которая позволит выиграть Пете при любой его игре, необходимо предотвратить выигрыш Пети первым ходом.

Поэтому воспользуемся формулой «=D32+E32» в ячейке D33 (сумма камней первой и второй куч в ходе Пети) установим условное форматирование, что даёт возможность не только отследить, но и визуализировать выигрыш Пети первым ходом (рис. 15, 16).

Рис. 15

Рис. 16

Заключительным шагом решения является нахождение максимального значения S, при котором Ваня гарантированно одерживает победу.

Методом подбора определяем $S = 24$, при котором Ваня одерживает победу своим первым ходом при любой игре Пети (рис. 17).

Проверим, является ли найденное значение максимальным, как того требует условие задачи. Подставим значение $S = 25$, видим, что значение ячейки D45 свидетельствует нам о выигрыше Пети первым ходом, что противоречит условию задачи (рис. 18).

Рис. 17

Рис. 18

Следовательно, значение $S = 24$ удовлетворяет условию задачи и гарантирует выигрыш Вани при любой игре Пети.

Представленный метод является универсальным и позволяет решать задачи с одной, двумя и более кучами камней (при желании их количество может быть увеличено операцией копирования).

Использование табличного процессора позволяет оптимизировать процесс решения задач 19–21 КЕГЭ по информатике и может быть полезно обучающимся, не овладевшим навыками программирования. Предложенный способ решения позволяет визуализировать математическую модель, не только получить результаты, но и экспериментально проверить адекватность модели.

Второй способ решения задачи — использование языка программирования Python.

В основе решения лежит реализованный ранее рекурсивный перебор, который полностью моделирует аналитический способ решения.

1.1. Решение задачи № 19.

Оформим функцию $g(h)$, которая будет возвращать результат операции с камнями в каждой из куч согласно условию задачи.

Недостатком использования рекурсивной функции является многократные вызовы функции самой себя, это заметно замедляет обработку больших массивов данных, решить проблему позволяет использование операции мемоизации (кэширования) — запоминания, при котором многократно вычисляемые однотипные операции сохраняются в памяти и вызываются по мере надобности:

```
from functools import lru_cache
def g(h):
    x, y=h # Операция распаковки
    return (x*3, y), (x, y*3), (x+1, y), (x, y+1)
@ lru_cache (None)
```

1.2. Оформим функцию $f(h)$, которая определяет положение выигрывающего игрока согласно игровой стратегии. Для этого введем следующие обозначения:

v — количество камней, с которого начинается выигрыш (по условию задачи);
 $p1$ — позиция, при которой Петя выигрывает своим первым ходом;
 $v1$ — позиция, при которой Ваня выигрывает своим первым ходом;
 $p2$ — позиция, при которой Петя выигрывает своим вторым ходом;
 $v2$ — позиция, при которой Ваня выигрывает своим вторым ходом.

По условию задачи Петя ошибся своим первым ходом, необходимо найти минимальное количество камней, при котором Ваня выиграет первым ходом. Именно поэтому мы применяем для определения позиции $v1$ функцию $any(...)$, которая возвращает «истина» при любом значении, удовлетворяющем данному условию.

Дописываем условия выигрыша Пети вторым ходом и Вани вторым ходом.

Для позиции $v2$ используем функцию $all(...)$, которая возвращает «истина» при всех значениях, удовлетворяющем данному условию.

Для задачи №19 последние две строчки кода не обязательны, так как требуется найти только выигрыш Вани первым ходом.

Однако мы считаем, что обучающим надо представить сразу полноценный код, который реализует классический алгоритм и связывает единой логикой все три задачи:

```
def f(h):
    x, y=h
    if x+y >=79: return 'v' # Условие выигрыша
    if any (f(i)=='v' for i in g(h)): return 'p1' # Условие выигрыша Пети 1 ходом
    if any (f(i)=='p1' for i in g(h)): return 'v1' # Условие выигрыша Вани 1 ходом после неудачного хода Пети
    if any (f(i)=='v1' for i in g(h)): return 'p2' # Условие выигрыша Пети 2 ходом
    if all (f(i)=='p1' or f(i)=='p2' for i in g(h)): return 'v2' # Условие выигрыша Вани 2 ходом
```

1.3. Простой переборный цикл находит все значения, при которых Ваня выигрывает своим первым ходом при неудачном ходе Пети:

```
for i in range(1, 73):
    h = 6, i
    if f(i) == 'v1':
        print(i, f(i))
```

1.4. Полный текст программы представляет собой простой и понятный код:

```
from functools import lru_cache
def g(h):
    x, y = h
    return (x*3, y), (x, y*3), (x+1, y), (x, y+1)
@lru_cache(None)
def f(h):
    x, y = h
    if x+y >= 79: return 'v'
    if any(f(i) == 'v' for i in g(h)): return 'p1'
    if any(f(i) == 'p1' for i in g(h)): return 'v1'
    if any(f(i) == 'v1' for i in g(h)): return 'p2'
    if all(f(i) == 'p1' or f(i) == 'p2' for i in g(h)): return 'v2'
    for i in range(1, 73):
        h = 6, i
        if f(i) == 'v1':
            print(i, f(i))
```

Минимальное количество камней, при котором Ваня выигрывает первым ходом равно 9.

1.5. Решение задачи № 20.

Программа аналогична задаче 19. Меняются только две строчки кода. Это строка, определяющая позицию v1. Функция any(..) меняется на функцию all(..) (необходимое условие, исходя из определения проигрышной позиции).

И в цикле выводится условие победы Пети вторым ходом.

Полный тест программы выглядит следующим образом:

```
from functools import lru_cache
def g(h):
    x, y = h
    return (x*3, y), (x, y*3), (x+1, y), (x, y+1)
@lru_cache(None)
def f(h):
    x, y = h
    if x+y >= 79: return 'v'
    if any(f(i) == 'v' for i in g(h)): return 'p1'
    if all(f(i) == 'p1' for i in g(h)): return 'v1'
    if any(f(i) == 'v1' for i in g(h)): return 'p2'
    if all(f(i) == 'p1' or f(i) == 'p2' for i in g(h)): return 'v2'
    for i in range(1, 73):
        h = 6, i
        if f(i) == 'p2':
            print(i, f(i))
```

Значения, при которых Петя выигрывает своим вторым ходом, равны 8, 20, 23

1.6. Решение задачи № 21.

Программа аналогична задаче 20. Меняется одна строка кода. В цикле выводится условие победы Вани при любом ходе Пети. Полный тест программы:

```
from functools import lru_cache
def g(h):
    x, y=h
    return (x*3, y), (x, y*3), (x+1, y), (x, y+1)
@ lru_cache (None)
def f(h):
    x, y=h
    if x+y >=79: return 'v'
    if any (f(i)=='v' for i in g(h)): return 'p1'
    if all (f(i)=='p1' for i in g(h)): return 'v1'
    if any (f(i)=='v1' for i in g(h)): return 'p2'
    if all (f(i)=='p1' or f(i)=='p2' for i in g(h)): return 'v2'
    for i in range (1,73):
        h=6, i
        if f(i)=='v2':
            print(i, f(i))
```

Значение, при котором Ваня выиграет своим вторым ходом, равно 24.

Представленный метод решения является универсальным и позволяет решать задачи с любым количеством куч камней и глубиной рекурсии.

ВЫВОДЫ. Таким образом, описанные способы решения задач на игровые стратегии, представленные в контрольно-измерительных материалах компьютерного ЕГЭ по информатике, с использованием возможностей табличного процессора, языка программирования высокого уровня Python позволят учителям информатики общеобразовательных организаций оптимизировать процесс обучения школьников по решению задач на тему «Игровые стратегии», осуществлять дифференцированный подход в подготовке обучающихся к итоговой аттестации с учетом уровня освоения ими учебного материала, формировать способность выполнять обоснованный выбор программного обеспечения для решения задачи. Применение теоретических основ игровых стратегий и прикладных возможностей табличного процессора и инструментария программирования для построения компьютерной модели игры способствует формированию информационно-технологических компетенций у выпускников школы, позволяет им широко использовать средства информационных технологий для решения контекстных задач.

Литература

1. Алгоритмизация и основы программирования: 19-21: теория игр. URL: <https://kpolyakov.spb.ru/school/prog.htm> (дата обращения 18.03.2021).
2. Босова Л.Л. О курсе информатики в Российской школе // Известия Кыргызской академии образования. 2015. № 3 (35). С. 323-325. URL: https://www.elibrary.ru/download/elibrary_28360563_34496438.pdf (дата обращения: 12.12.2021).
3. Вабищевич С.В. МПИ как новый раздел педагогики // Методика преподавания информатики. URL: <https://phys.bspu.by/static/um/inf/mpi/lekc/lek2.pdf> (дата обращения 12.12.2021).
4. Горбулина Т.С. Теория игр. Задача № 26 ЕГЭ по информатике и ИКТ / Т.С. Горбулина, К.А. Колесниченко. Курск, 2020. URL: https://www.kiro46.ru/docs/EGE_Informatika_01.pdf (дата обращения: 12.12.2021).
5. Григорьева Н.Д. Теория игр и ее практическое применение: учебные материалы / Н.Д. Григорьева. URL: <https://works.doklad.ru/view/BMUfSH7EWYU.html> (дата обращения: 12.12.2021).

6. Губко М.В. Теория игр в управлении организационными системами / М.В. Губко, Д.А. Новиков. М.: Изд-во МФТИ, 2005. 138 с.
7. Дмитриева Ю.Ю. История развития и формирование теории игр / Ю.Ю. Дмитриева, В.С. Келий. — Текст: электронный // Студенческий научный форум — 2014: международная студенческая научная конференция, 15 февраля-31 марта 2014 г., г. Пенза. Саратов: Академия Естествознания, 2014. URL: <https://scienceforum.ru/2014/article/2014001981> (дата обращения: 12.12.2021).
8. Костромин А.В. Теория игр: конспект лекций / А.В. Костромин, Д.М. Мухаметгалеев; Казанский федеральный университет. Казань, 2013. 87 с. URL: http://libweb.kpfu.ru/ebooks/72-IEF/72_198_A5kl-000487.pdf (дата обращения: 12.12.2021).
9. Мажитова М.Р. Формирование ИКТ-компетенций в преподавании информатики и ИКТ / М.Р. Мажитова // Информатика. Все для учителя. 2017. № 9 (81). С. 2–4.
10. Мулен Э. Теория игр с примерами из математической экономики / Э. Мулен. М.: Мир, 1985. URL: <http://www.pseudology.org/lokhotron/GamesTheory.pdf> (дата обращения: 12.12.2021).
11. Нейман Дж. Фон. Теория игр и экономическое поведение / Дж. Фон Нейман, О. Моргенштерн. М.: Наука, 1970. 983 с.
12. Петросян Л.А. Теория игр / Л.А. Петросян, Н.А. Зенкенвич, Е.В. Шевкопляс. СПб, 2012. URL: http://static.ozone.ru/multimedia/book_file/1005735639.pdf (дата обращения: 12.12.2021).
13. Потешкина Г.В. Современные проблемы в преподавании информатики / Г.В. Потешкина. URL: <https://cyberleninka.ru/article/n/sovremennye-problemy-v-prepodavanii-informatiki> (дата обращения: 12.12.2021).
14. Универсальные компетентности и новая грамотность: от лозунгов к реальности / под редакцией: М.С. Добряковой, И.Д. Фрумина; Национальный исследовательский университет «Высшая школа экономики». М: Высшая школа экономики, 2020. 472 с.
15. ФИПИ. Демоверсии, кодификаторы, спецификации (ЕГЭ). URL: <http://fipi.ru/ege-i-gve-11/demoversii-specifikacii-kodifikatory> (дата обращения: 12.12.2021).

References

1. *Algoritimizaciya i osnovy programmirovaniya: 19–21: teoriya igr*. [Algorithmization and fundamentals of programming: 19–21: Game theory]. URL: <https://kpolyakov.spb.ru/school/prog.htm> (data obrashcheniya: 03.18.2021). (In Russian).
2. Bosova L.L. *O kurse informatiki v Rossiiskoi shkole* [About the informatics course at the Russian School] // *Izvestiya Kyrgyzskoi akademii obrazovaniya*. 2015. № 3 (35). S. 323–325. (In Russian) URL: https://www.elibrary.ru/download/elibrary_28360563_34496438.pdf (data obrashcheniya: 12.12.2021). (In Russian).
3. *Vabishchevich S.V. MPI kak novyi razdel pedagogiki* [MPI as a new section of pedagogy] // *Metodika prepodavaniya informatiki*. / S.V. Vabishchevich // *Metodika prepodavaniya informatiki*. URL: <https://phys.bspu.by/static/um/inf/mpi/lekc/lek2.pdf> (data obrashcheniya: 12.12.2021). (In Russian).
4. Gorbulina T.S. *Teoriya igr. Zadacha № 26 EGEH po informatike i IKT* [Game theory. Task number 26 of the exam in informatics and ICT] / T.S. Gorbulina, K.A. Kolesnichenko Kursk, 2020. URL: https://www.kiro46.ru/docs/EGE_Informatika_01.pdf (data obrashcheniya: 12.12.2021). (In Russian).
5. Grigor'eva N.D. *Teoriya igr i ee prakticheskoe primeneniye: uchebnye ma-terialy* [Game theory and its practical application: teaching materials] / N.D. Grigor'eva. — URL: <https://works.doklad.ru/view/BMufSH7EWYU.html> (data obrashcheniya: 12.12.2021) (In Russian)
6. Gubko M.V. *Teoriya igr v upravlenii organizatsionnymi sistemami* [Game theory in the management of organizational systems] / M.V. Gubko, D.A. Novikov. M: MFTU, 2005. 138 s. (In Russian).
7. Dmitrieva YU. *Istoriya razvitiya i formirovaniye teorii igr* [History of development and formation of game theory] // *Studencheskii nauchnyi forum 2014: mezhdunarodnaya studencheskaya nauchnaya konferentsiya*, 15 february-31 march 2014 g., g. Penza. Saratov: Akademiya Estestvoznaniya, 2014. URL: <https://scienceforum.ru/2014/article/2014001981> (data obrashcheniya: 12.12.2021). (In Russian).

8. Kostromin A.V. *Teoriya igr: konspekt lektsii* [Game Theory: Lecture Notes] / A.V. Kostromin, D.M. Mukhametgaleev; Kazanskii federal'nyi universitet. Kazan', 2013. 87 s. URL: http://libweb.kpfu.ru/ebooks/72-IEF/72_198_A5kl-000487.pdf (data obrashcheniya: 12.12.2021). (In Russian).
9. Mazhitova M.R. *Formirovanie IKT-kompetentsii v prepodavanii in-formatiki i IKT* [Formation of ICT competence in teaching computer science and ICT] / M.R. Mazhitova. Tekst: neposredstvennyi // Informatika. Vse dlya uchitelya., 2017. № 9 (81). S. 2–4. (In Russian).
10. Mulen E.H. *Teoriya igr s primerami iz matematicheskoi ehkonomiki* [Game theory with examples from mathematical Economics] / E.H. Mu-len. Moskva: Mir, 1985. URL: <http://www.pseudology.org/lokhotron/GamesTheory.pdf> (data obrashcheniya: 12.12.2021). (In Russian).
11. Nejman Dzh. *Fon Teorija igr i jekonomicheskoe povedenie* [Game theory and economic behavior] / Dzh. Fon Nejman, O.Mongernshtern. M.: Nauka, 1970. 983 s. (In Russian).
12. Petrosyan L.A. *Teoriya igr* [Game theory] / L.A. Petrosyan, N.A. Zenkovich, E.V. Shevkoplyas. SPb, 2012. URL: http://static.ozone.ru/multimedia/book_file/1005735639.pdf (data obrashcheniya: 12.12.2021). (In Russian).
13. Poteshkina G.V. *Sovremennye problemy v prepodavanii informatiki* [Modern problems in teaching computer science] / G.V. Poteshkina. URL: <https://cyberleninka.ru/article/n/sovremennye-problemy-v-prepodavanii-informatiki> (data obrashcheniya: 12.12.2021). (In Russian).
14. *Universal'nye kompetentnosti i novaya gramotnost': ot lozungov k real'-nosti* [Universal competencies and new literacy: from slogans to reality] / pod redaktsiei: M.S. Dobryakovoi, I.D. Frumina; Natsional'nyi is-sledovatel'skii universitet «Vysshaya shkola ehkonomiki». M.: High School of Economics, 2020. 472 s. (In Russian).
15. *FIPI. Demoversii, kodifikatory, spetsifikatsii (EGEH)* [FIPI. Demos, codifications, specifications (USE)]. URL: <http://fipi.ru/ege-i-gve-11/demoversii-spezifikacii-kodifikatory> (data obrashcheniya: 12.12.2021). (In Russian).