

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**



**ОЛИМПИАДА ШКОЛЬНИКОВ
«ГРАНИТ НАУКИ»**

ИНФОРМАТИКА

**Методические указания для подготовки к заключительному туру олимпиады
в 2023/2024 учебном году
для 10-11 классов**

Санкт-Петербург, 2024



ОГЛАВЛЕНИЕ

Информатика.....	3
Задания для подготовки к олимпиаде.....	6
Список рекомендуемой литературы для подготовки	21



Информатика

Характер и уровень сложности олимпиадных задач направлены на достижение целей проведения олимпиады: выявить способных участников, твердо владеющих школьной программой и наиболее подготовленных к освоению образовательных программ технических ВУЗов, обладающих логикой и творческим характером мышления, умеющих алгоритмически описать реальные ситуации из различных предметных областей и применить к ним наиболее подходящие методы информатики.

Задания дифференцированы по сложности и требуют различных временных затрат на верное и полное решение. Они охватывают все разделы школьной программы, но носят, в большинстве, комплексный характер, позволяющий варьировать оценки в зависимости от проявленных в решении творческих подходов и продемонстрированных технических навыков. Участники должны самостоятельно разбить задачу на подзадачи, грамотно выполнить решение каждой подзадачи, синтезировать решение всей задачи из решений отдельных подзадач. Успешное выполнение олимпиадной работы не предполагает знаний, выходящих за пределы школьной программы, но требует творческого подхода, логического мышления, умения увидеть и составить правильный и оптимальный план решения, четкого и технически грамотного выполнения каждой части решения.

Очный тур олимпиады проводится только в письменной форме. Каждый участник олимпиады получает билет, содержащий шесть заданий. При выполнении заданий требуется дать теоретическое обоснование и привести все этапы решения задачи:

- при определении результата перевода заданного числа из одной системы счисления в другую следует указать правило перевода в общем виде, а затем привести схему перевода заданного числа в указанную систему счисления;
- при определении значений ячеек электронной таблицы приводятся результаты, формулы, полученные при копировании, даются пояснения о типах ссылок в формулах, как типы ссылок влияют на результат и др.;
- при ответе на вопрос об объемах памяти, выделяемой для кодирования символьной информации, необходимо указать особенности системы кодирования, привести подробные расчеты;
- при ответе на вопрос о поисковых запросах следует привести результаты промежуточных расчетов, полученных после выполнения запросов;
- при ответе на вопрос об определении значения переменной после выполнения программы, алгоритм которой приводится в билете, следует пояснить работу каждого оператора (блока), выписать промежуточные и окончательный результаты;
- необходимо дать развернутый ответ с пояснениями при составлении расширенного алгоритма, пояснить алгоритм графически;
- ответы на задачи, посвященные темам, связанным с сетями, IP-адресами, масками подсети должны сопровождаться публикацией всей информации без сокращений;
- в случае решения задач, связанных с таблицами истинности необходимо описывать все промежуточные результаты;
- при решении задачи на программирование необходимо составить блок-схему или описать алгоритм решения задачи, написать текст программы на любом из языков программирования, указав используемый язык программирования и его версию, допускается использовать только стандартную библиотеку языка;



– алгоритм должен раскрывать решение на каждом шаге. Код программы должен состоять из простых и понятных команд. Запрещено использовать уникальные особенности языка программирования (классы, методы, объекты, попарный обмен между переменными, специальные функции и библиотеки и др.) для реализации решения. Считывание заранее рассчитанных значений не засчитывается за решение задачи. Код должен содержать достаточное для понимания логики работы количество комментариев;

– программы, составленные участниками, тестируются и всесторонне оцениваются с учетом быстродействия, рациональности, минимального использования памяти компьютера и оригинальности предложенных решений;

– при использовании методов научного перебора необходимо привести доказательства использования этого метода.

При подготовке к олимпиаде следует повторить приведенные ниже разделы.

1. Информация и информационные процессы

1.1. Информация и ее кодирование 1.2. Виды информационных процессов

1.3. Процесс передачи информации, источник и приемник информации. Сигнал, кодирование и декодирование.

1.4. Искажение информации

1.5. Дискретное (цифровое) представление текстовой, графической, звуковой информации и видеоинформации. Единицы измерения количества информации

1.6. Скорость передачи информации

1.7. Системы, компоненты, состояние и взаимодействие компонентов. Информационное взаимодействие в системе, управление, обратная связь

1.8. Моделирование

1.9. Описание (информационная модель) реального объекта и процесса, соответствие описания объекту и целям описания. Схемы, таблицы, графики, формулы как описания

1.10. Математические модели

1.11. Использование сред имитационного моделирования (виртуальных лабораторий) для проведения компьютерного эксперимента в учебной деятельности

1.12. Системы счисления

1.13. Позиционные системы счисления

1.14. Двоичное представление информации

1.15. Логика и алгоритмы

1.16. Высказывания, логические операции, кванторы, истинность высказывания

1.17. Цепочки (конечные последовательности), деревья, списки, графы, матрицы (массивы), псевдослучайные последовательности

1.18. Индуктивное определение объектов

1.19. Вычислимые функции, полнота формализации понятия вычислимости, универсальная вычислимая функция

1.20. Кодирование с исправлением ошибок

1.21. Сортировка

1.22. Элементы теории алгоритмов

1.23. Формализация понятия алгоритма

1.24. Вычислимость. Эквивалентность алгоритмических моделей

1.25. Построение алгоритмов и практические вычисления

1.26. Языки программирования



- 1.27. Типы данных
- 1.28. Основные конструкции языка программирования. Система программирования
- 1.29. Основные этапы разработки программ. Разбиение задачи на подзадачи
- 2. Информационная деятельность человека**
 - 2.1. Профессиональная информационная деятельность. Информационные ресурсы
 - 2.2. Экономика информационной сферы
 - 2.3. Информационная этика и право, информационная безопасность=
- 3. Средства ИКТ**
 - 3.1. Архитектура компьютеров и компьютерных сетей
 - 3.2. Программная и аппаратная организация компьютеров и компьютерных систем. Виды программного обеспечения
 - 3.3. Операционные системы. Понятие о системном администрировании
 - 3.4. Безопасность, гигиена, эргономика, ресурсосбережение, технологические требования при эксплуатации компьютерного рабочего места
 - 3.5. Технологии создания и обработки текстовой информации
 - 3.6. Понятие о настольных издательских системах. Создание компьютерных публикаций
 - 3.7. Использование готовых и создание собственных шаблонов. Использование систем проверки орфографии и грамматики. Тезаурусы. Использование систем двуязычного перевода и электронных словарей
 - 3.8. Использование специализированных средств редактирования математических текстов и графического представления математических объектов
 - 3.9. Использование систем распознавания текстов
 - 3.10. Технология создания и обработки графической и мультимедийной информации
 - 3.11. Форматы графических и звуковых объектов
 - 3.12. Ввод и обработка графических объектов
 - 3.13. Ввод и обработка звуковых объектов
 - 3.14. Обработка числовой информации
 - 3.15. Обработка изображений
 - 3.16. Математическая обработка статистических данных
 - 3.17. Использование динамических (электронных) таблиц для выполнения учебных заданий из различных предметных областей
 - 3.18. Использование инструментов решения статистических и расчетно-графических задач
 - 3.19. Технологии поиска и хранения информации
 - 3.20. Системы управления базами данных. Организация баз данных
 - 3.21. Использование инструментов поисковых систем (формирование запросов)
 - 3.22. Телекоммуникационные технологии
 - 3.23. Специальное программное обеспечение средств телекоммуникационных технологий
 - 3.24. Инструменты создания информационных объектов для Интернет

**Задания для подготовки к олимпиаде**

Задача 1. 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 (по своему выбору) один камень или увеличить количество камней в куче в два раза. Например, пусть в одной куче 10 камней, а в другой 7 камней; такую позицию в игре будем обозначать $(10, 7)$. Тогда за один ход можно получить любую из четырёх позиций: $(11, 7)$, $(20, 7)$, $(10, 8)$, $(10, 14)$. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда суммарное количество камней в кучах становится не менее 73. Победителем считается игрок, сделавший последний ход, т.е. первым получивший такую позицию, что в кучах всего будет 73 камня или больше.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. Например, при начальных позициях $(6, 34)$, $(7, 33)$, $(9, 32)$ выигрышная стратегия есть у Пети. Чтобы выиграть, ему достаточно удвоить количество камней во второй куче.

Задание 1. Для каждой из начальных позиций $(6, 33)$, $(8, 32)$ укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 2. Для каждой из начальных позиций $(6, 32)$, $(7, 32)$, $(8, 31)$ укажите, кто из игроков имеет выигрышную стратегию. В каждом случае опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии.

Задание 3. Для начальной позиции $(7, 31)$ укажите, кто из игроков имеет выигрышную стратегию. Опишите выигрышную стратегию; объясните, почему эта стратегия ведёт к выигрышу, и укажите, какое наибольшее количество ходов может потребоваться победителю для выигрыша при этой стратегии. Постройте дерево всех партий, возможных при указанной Вами выигрышной стратегии. Представьте дерево в виде рисунка или таблицы.

Ответ:

1. Ваня первым ходом.
2. Петя вторым ходом.
3. Ваня вторым ходом.

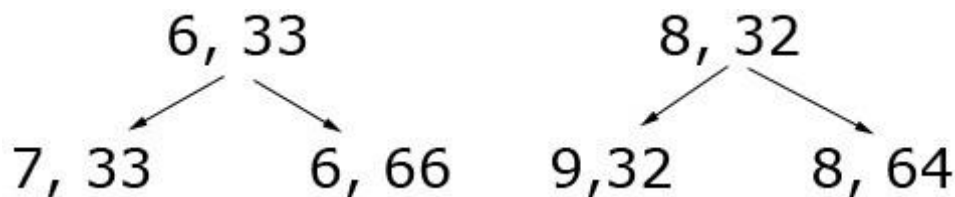
Решение:

Задание 1. Кто из игроков имеет выигрышную стратегию при позициях $(6, 33)$ и $(8, 32)$?

Игроки могут делать два хода — прибавить к любой из куч один камень, или умножить количество камней в любой из куч на два. Игрок выигрывает, когда суммарное количество камней в куче становится ≥ 73 . Будем считать, что прибавление одного камня в маленькую кучу — самый слабый ход, а увеличение в два раза большой кучи — самый сильный ход. Остальные ходы нам не интересны.



Рассмотрим это на графе:



Петя первым ходом не может получить суммарное количество ≥ 73 , зато Ваня может, удвоив количество камней во второй куче после хода Пети:

$$7 + 2 \cdot 33 = 73$$

$$6 + 2 \cdot 66 = 138$$

$$9 + 2 \cdot 32 = 73$$

$$8 + 2 \cdot 64 = 136$$

То есть при позициях (6, 33) и (8, 32) второй игрок (Ваня) выигрывает первым ходом.

Так как Петя, получив такие позиции, проиграл, будем считать, что позиции (6, 33) и (8, 32) проигрышные, и игрок, которому они достанутся, проиграет.

Задание 2. Кто из игроков имеет выигрышную стратегию при позициях (6, 32), (7, 32), (8, 31)?

Петя ходит первым, очевидно, что он хочет выиграть, а для этого должен играть правильно. То есть он должен стараться сделать так, чтобы Ване досталась проигрышная позиция. Мы знаем две проигрышные позиции из пункта 1 — (6, 33) и (8, 32). Эти проигрышные позиции Петя может сделать своим первым ходом для Вани из предложенных позиций:

$$(6, 32) \rightarrow (6, 33)$$

$$(7, 32) \rightarrow (8, 32)$$

$$(8, 31) \rightarrow (8, 32)$$

То есть Петя сделал для Вани проигрышную позицию, в результате чего Ваня проиграет, а Петя выиграет своим вторым ходом. Будем считать, что позиции (6, 32), (7, 32), (8, 31) выигрышные, так как они достались Пете и Петя выиграл.

Задание 3. Кто из игроков имеет выигрышную стратегию при позиции (7, 31)?

Рассмотрим все возможные ходы Пети из этой позиции: (7, 31)
 $\rightarrow (8, 31)$

$$(7, 31) \rightarrow (7, 32)$$

$$(7, 31) \rightarrow (14, 31)$$

$$(7, 31) \rightarrow (7, 62)$$

То есть Ване могут достаться позиции (8, 31), (7, 32), (14, 31), (7, 62).

Рассмотрим каждую:

(8, 31) — выигрышная позиция, исходя из пункта 2 решения; (7, 32)

— выигрышная позиция, исходя из пункта 2 решения; (14, 31) — Ваня выиграет, умножив 31 на 2;

(7, 62) — Ваня выиграет, умножив 62 на 2.

Выходит, что все позиции, которые достались Ване — выигрышные. При этом при позициях (8, 31) и (7, 32) Ване потребуется сделать два хода для победы. То есть при начальной позиции (7, 31) Ваня выиграет максимум двумя ходами.



Задача 2. В электронной таблице представлена информация о продуктах. В первой таблице указано количество продуктов на складе на складе (в кг.), во второй – стоимость.

	A	B	C	D	E	F
1	Продукт	Молоко	Сливки	Сыр	Творог	Сметана
2	Количество	55	77	100	150	70
3						
4						
5	Продукт	Сметана	Сыр	Молоко	Сливки	Творог
6	Цена	70	350	60	65	72

Заполните столбец «Цена» первой таблицы данными о стоимости из второй. Запишите формулы для вычислений и получившиеся результаты.

Решение:

8	Продукт	Молоко	Сливки	Сыр	Творог	Сметана	=ГПР(B1;\$B\$5:\$F\$6;2)
9	Количество	55	77	100	150	70	=ГПР(C1;\$B\$5:\$F\$6;2)
10	Цена	60	65	65	72	65	=ГПР(D1;\$B\$5:\$F\$6;2)
11							=ГПР(E1;\$B\$5:\$F\$6;2)
12							=ГПР(F1;\$B\$5:\$F\$6;2)
13							

Задача 3. С тех пор, как Иван изучил формулы в Excel, он ежедневно оттачивает свое мастерство. Сегодня он записал цифру 2 в ячейку A1, в соседнюю ячейку B1 он внес формулу =ОКРУГЛ(ОСТАТ(8;\$A\$1)+СТЕПЕНЬ(A1;2)-ЕСЛИ(A1>0;A1^2/3;A1/3);1) и растянул до ячейки F1. На следующей строке, начиная с ячейки B1 Иван ввел формулу =ОКРУГЛ(ОСТАТ(B1;A1)-КОРЕНЬ(ABS(A1-B1))+ЕСЛИ(A1+B1<2;B1/A1;A1+3);2) и скопировал в диапазон B2:F2. На следующей строке он ввел первой цифру 2,5 и повторил действия, как для первой строки. А в четвертую (диапазон A4:F4) скопировал формулу со 2 строки. В пятой строке Иван ввел первой цифру 3 и снова решил повторить действия, как и для строки 1. В ячейку A6 он вставил формулу из A2 и растянул до ячейки F6. После этих действий Иван решил воспользоваться инструментом условного форматирования, выставив следующие условия:

- значения больше \$D\$1 выделить красным цветом;
- значения меньше \$C\$3 выделить желтым цветом;
- значения между 10 и 16 выделить зеленым цветом.

Укажите, какое количество ячеек оказалось выделено желтым цветом?

Ответ: 15

Решение:

	A	B	C	D	E	F
1	2	2,7	4,9	16	170,7	19425,7
2	4,86	6,42	5,87	17,26	171,54	19289,32
3	2,5	4,2	11,8	92,8	5741,2	21974252
4	5,9	7,84	16	101,04	3736,34	21969567
5	3	6	24	384	98304	6,44E+09
6	4,27	4,76	8,03	74,08	18042,73	6,44E+09



Фрагмент в режиме отображения формул:

H2		
		fx
	A	B
1	2	=ОКРУГЛ(ОСТАТ(8;\$A\$1)+СТЕПЕНЬ(A1;2)-ЕСЛИ(A1>0;A1^2/3;A1/3);1)
2	=ОКРУГЛ(ОСТАТ(B1;A1)-КОРЕНЬ(ABS(A1-B1))+ЕСЛИ(A1+B1<2;B1/A1;A1+3);2)	=ОКРУГЛ(ОСТАТ(C1;B1)-КОРЕНЬ(ABS(B1-C1))+ЕСЛИ(B1+C1<2;C1/B1;B1+3);2)
3	2,5	=ОКРУГЛ(ОСТАТ(8;\$A\$1)+СТЕПЕНЬ(A3;2)-ЕСЛИ(A3>0;A3^2/3;A3/3);1)
4	=ОКРУГЛ(ОСТАТ(B3;A3)-КОРЕНЬ(ABS(A3-B3))+ЕСЛИ(A3+B3<2;B3/A3;A3+3);2)	=ОКРУГЛ(ОСТАТ(C3;B3)-КОРЕНЬ(ABS(B3-C3))+ЕСЛИ(B3+C3<2;C3/B3;B3+3);2)
5	3	=ОКРУГЛ(ОСТАТ(8;\$A\$1)+СТЕПЕНЬ(A5;2)-ЕСЛИ(A5>0;A5^2/3;A5/3);1)
6	=ОКРУГЛ(ОСТАТ(B5;A5)-КОРЕНЬ(ABS(A5-B5))+ЕСЛИ(A5+B5<2;B5/A5;A5+3);2)	=ОКРУГЛ(ОСТАТ(C5;B5)-КОРЕНЬ(ABS(B5-C5))+ЕСЛИ(B5+C5<2;C5/B5;B5+3);2)
7		
8		

Задача 4.

Для получения секретной информации нашим разведчикам необходимо вскрыть сейф, замок которого состоит из 5 ячеек (B2:F2). Задача осложняется тем, что код можно ввести только один раз (метод подбора не поможет), в ячейках может находиться как числовая, так и текстовая информация, и самое главное, формулы с подсказками из ячеек B4:B12 были скопированы в ячейки C5:C13, что привело к смещению ссылок.

Помогите разведчикам найти код и выиграть эту схватку. В Вашем распоряжении значения формул с подсказками в первоначальном состоянии (B4:B12) и формулы в режиме отображений формул после операции копирования (C5:C13).

	A	B	C	D	E	F	G	H	I	J
1										
2	Код:									
3										

	A	B
4	№1	0
5	№2	4
6	№3	3
7	№4	#ДЕЛ/0!
8	№5	#ЗНАЧ!
9	№6	дбор
10	№7	3
11	№8	4
12	№9	17
13		

	C
4	
5	=СЧЁТЕСЛИ(C3:G3;">=10")+СЧЁТЕСЛИ(C3:G3;"<=0")
6	=СЧЁТ(C3:G3)
7	=ДЛСТР(ЕСЛИ(ЕТЕКСТ(D3);D3;ТЕКСТ(D3;"#")))+ДЛСТР(ЕСЛИ(ЕТЕКСТ(E3);E3;ТЕКСТ(E3;"#")))
8	=СУММ(C3:E3)/(F\$2-G\$2)
9	=\$D\$2+F\$2+G\$2
10	=ПСТР("Подбор";\$F3;6)
11	=ПОИСК(СЦЕПИТЬ(ЕСЛИ(ЕТЕКСТ(D3);D3;"");ЕСЛИ(ЕТЕКСТ(E3);E3;"");"гипнотизер"))
12	=ДЛСТР(СЦЕПИТЬ(ЕСЛИ(ЕТЕКСТ(D3);D3;"");ЕСЛИ(ЕТЕКСТ(E3);E3;"")))+С3
13	=СУММ(C3:G3)

Решение.

1. Находим первоначальное значение формул с учетом операции копирования.



	A	B
4	№1	=СЧЁТЕСЛИ(B2:F2;">=10")+СЧЁТЕСЛИ(B2:F2;"<=0")
5	№2	=СЧЁТ(B2:F2)
6	№3	=ДЛСТР(ЕСЛИ(ЕТЕКСТ(C2);C2;ТЕКСТ(C2;"#")))+ДЛСТР(ЕСЛИ(ЕТЕКСТ(D2);D2;ТЕКСТ(D2;"#")))
7	№4	=СУММ(B2:D2)/(E\$2-F\$2)
8	№5	=\$D\$2+E\$2+F\$2
9	№6	=ПСТР("Подбор";\$F2;6)
10	№7	=ПОИСК(СЦЕПИТЬ(ЕСЛИ(ЕТЕКСТ(C2);C2;"");ЕСЛИ(ЕТЕКСТ(D2);D2;"");"гипнотизер"))
11	№8	=ДЛСТР(СЦЕПИТЬ(ЕСЛИ(ЕТЕКСТ(C2);C2;"");ЕСЛИ(ЕТЕКСТ(D2);D2;"")))+B2
12	№9	=СУММ(B2:F2)

2. По формуле №1, находящейся в ячейке B4 «=СЧЁТЕСЛИ(B2:F2;">=10")+СЧЁТЕСЛИ(B2:F2;"<=0")» равной 0 определяем, что все числовые значения находятся в диапазоне от 1 до 9.

3. По формуле №2, находящейся в ячейке B5 «=СЧЁТ(B2:F2)» равной 4 определяем, что количество числовых значений равно 4, следовательно одна из ячеек содержит текстовую информацию.

4. Формула №3, находящаяся в ячейке B6 «=ДЛСТР(ЕСЛИ(ЕТЕКСТ(C2);C2;ТЕКСТ(C2;"#")))+ДЛСТР(ЕСЛИ(ЕТЕКСТ(D2);D2;ТЕКСТ(D2;"#"))))» равная 3 подсчитывает количество символов в ячейках C2 и D2, переводя числовые значения в текстовые. В соответствии с тем, что числовые значения находятся в диапазоне от 1 до 9, сумма количества символов, при условии нахождения в этих ячейках числовых значений, равнялась бы 2. Значение 3 говорит о том, что одна из ячеек имеет длину 2 символа и это – ячейка с текстовой информацией.

5. Значение формулы №4, находящейся в ячейке B7 «=СУММ(B2:D2)/(E\$2-F\$2)» является сообщением об ошибке деления на 0 «#ДЕЛ/0!», это возможно при условии, что значения в ячейках E2 и F2 одинаковы.

6. Значение формулы №5, находящейся в ячейке B8 «=\$D\$2+E\$2+F\$2» является сообщением об ошибке «#ЗНАЧ!» означающее, что значение используемое в формуле, имеет неправильный тип данных. Это возможно в том случае, если в этих ячейках находится текстовая информация. В соответствии с пунктом 4 данного решения мы знаем, что текст находится либо в ячейке C2 либо D2, следовательно ячейка с текстовой информацией – D2.

7. Значение формулы №6, находящейся в ячейке B9 «=ПСТР("Подбор";\$F2;6)» равно подстроке «дбор», откуда F2 равно 3, а значения в ячейках E2 и F2 одинаковы, в соответствии с пунктом 5 данного решения. Следовательно, E2=F2=3.

8. Формула №7, находящаяся в ячейке B10 «=ПОИСК(СЦЕПИТЬ(ЕСЛИ(ЕТЕКСТ(C2);C2;"");ЕСЛИ(ЕТЕКСТ(D2);D2;"");"гипнотизер"))» показывает, что наше текстовое поле имеется в слове «гипнотизер» начиная с третьей позиции. Зная, что его длина 2 символа, в соответствии с пунктом 4 решения, получаем значение ячейки D2 = «пн».

9. Формула №8, находящаяся в ячейке B11 «=ДЛСТР(СЦЕПИТЬ(ЕСЛИ(ЕТЕКСТ(C2);C2;"");ЕСЛИ(ЕТЕКСТ(D2);D2;"")))+B2» равная 4, позволяет найти значение ячейки B2, как разницы между значением формулы и длиной текстового поля ячейки D2. B2=4-2=2.

10. Формула №9, находящаяся в ячейке B12 «=СУММ(B2:F2)» позволяет найти значение последней неизвестной ячейки C2 = 17-2-3-3 = 9.

Ответ:

B2	C2	D2	E2	F2
2	9	пн	3	3



Задача 5. Символом F обозначен результат логического выражения от 3 аргументов X, Y, Z , а $F1$ и $F2$ – обозначение промежуточных результатов:

X	Y	Z	$F1$	$F2$	F
0	0	0	0	1	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	0
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	1

Необходимо восстановить выражение по таблице истинности.

Ответ: $(x \wedge y) \rightarrow z \leftrightarrow x$

Решение:

- 1). Первое действие (результат $F1$) удовлетворяет условиям конъюнкции
- 2). Второй столбец соответствует результату следования первого действия из z .
- 3). Результирующий столбец F является результатом равносильности второго действия и x .

В результате получается: $F = (x \wedge y) \rightarrow z \leftrightarrow x$

Задача 6

Аня увлеклась изучением информатики. Она упростила выражение $F = \overline{\overline{x} \wedge y} \wedge (\overline{x \rightarrow y}) \vee (x \vee y) \wedge (x \vee z) \vee x \vee (\overline{x \wedge y})$, следуя основным законам математической логики, построила таблицу истинности и логическую схему. При этом результирующий столбец представила в виде последовательности четырех чисел в десятичной системе счисления. Например, если результирующий столбец будет 10001101_2 , то в десятичной системе последовательность будет иметь вид 2031_{10} (так как $10_2 = 2_{10}$, $00_2 = 0_{10}$ и т.д.). Напишите, какой ответ получила Аня в виде последовательности из четырех десятичных чисел.

Решение:

Выражение $F = \overline{\overline{x} \wedge y} \wedge (\overline{x \rightarrow y}) \vee (x \vee y) \wedge (x \vee z) \vee x \vee (\overline{x \wedge y})$ можно упростить следующим образом:

1. $\overline{\overline{x} \wedge y} \equiv \overline{\overline{x}} \vee \overline{\overline{y}} \equiv x \vee y$ по закону отрицания конъюнкции (или Де Моргана), а далее $\overline{\overline{x} \wedge y} \equiv x \wedge \overline{y}$ по закону двойного отрицания (или инверсии);
2. $(x \rightarrow y) \equiv x \wedge \overline{y}$ по закону отрицания импликации;
3. $(x \vee y) \wedge (x \vee z) \equiv (y \wedge z) \vee x$ по закону дистрибутивности;
4. $x \vee (\overline{x \wedge y}) \equiv x \vee y$ по закону ортогонализации.

Итоговое выражение имеет вид: $F = (x \vee \overline{y}) \wedge (x \wedge \overline{y}) \vee (x \vee (y \wedge z)) \vee (x \vee y)$. Первым шагом будет расстановка приоритета действий:

1. Отрицание y .
2. Дизъюнкция в первых скобках
3. Конъюнкция во вторых скобках.
4. Дизъюнкция в третьих скобках.
5. Дизъюнкция в четвертых скобках.
6. Дизъюнкция действия 4 и x .
7. Конъюнкция действия 2 и 3



8. Дизъюнкция действий 7 и 6.

9. Дизъюнкция действий 8 и 5.

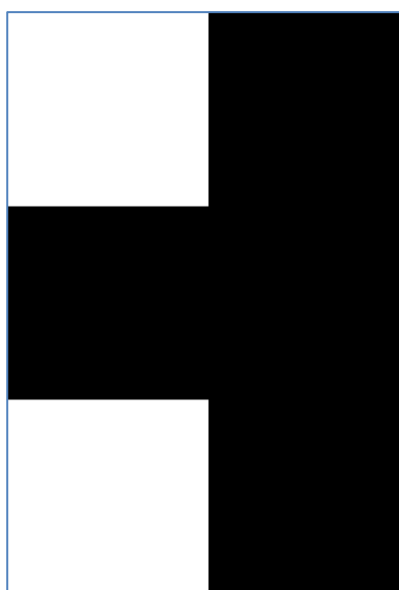
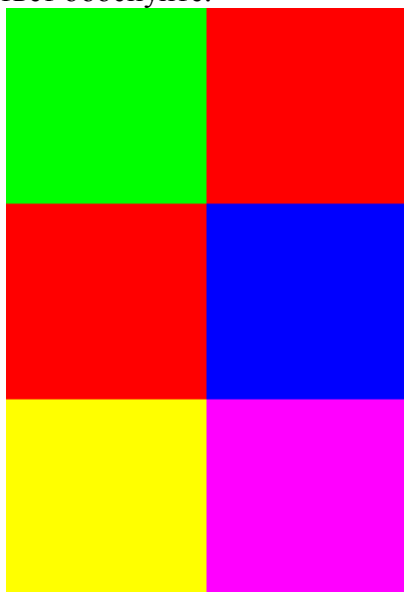
На основании данного приоритета операций на втором шаге строится таблица истинности:

x	y	Z	1	2	3	4	5	6	7	8	9
0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0	0	0	0	0
0	1	0	0	0	0	0	1	0	0	0	1
0	1	1	0	0	0	1	1	1	0	1	1
1	0	0	1	1	1	0	1	1	1	1	1
1	0	1	1	1	1	0	1	1	1	1	1
1	1	0	0	1	0	0	1	1	1	1	1
1	1	1	0	1	0	1	1	1	1	1	1

Для того, чтобы представить ответ в виде четырех десятичных чисел необходимо разбить полученный ответ в двоичной системе на четыре двоичных числа: $0_2, 11_2, 11_2, 11_2$, что в десятичной системе счисления будет выглядеть, как: $0_{10}, 3_{10}, 3_{10}, 3_{10}$.

Задача 7.

Цветное изображение, пиксели которого закодированы с помощью цветовой модели RGB, могут быть представлены в виде яркостных представлений отдельных каналов. На рисунке 1 представлено исходное изображение, а на рисунке 2 его яркостное представление по каналу G. Схематично изобразите яркостное представление каналов R и B, ответ обоснуйте.



Решение:

Из рисунка 2 делаем вывод, что максимальное значение яркостив канале зелёного цвета G представлено белым цветом, а минимальное чёрным цветом. Значит для канала яркости красного цвета R будут следующие соотношения:

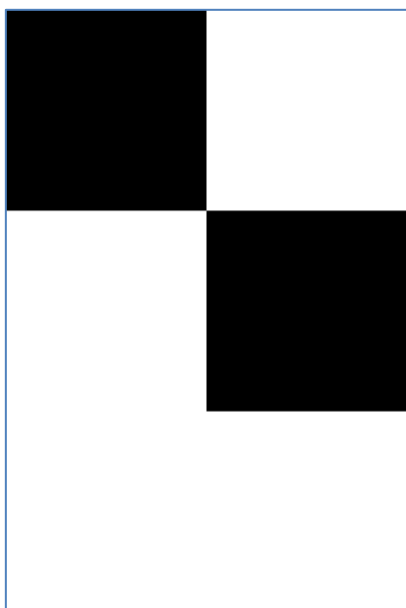
Зелёный – чёрный

Красный – белый

Синий – чёрный

Жёлтый – белый

Лиловый – белый



А для канала яркости синего цвета В будут следующие соотношения:

Зелёный – чёрный

Красный – чёрный

Синий – белый

Жёлтый – чёрный

Лиловый – белый



Задача 8.

ГИРЛЯНДА

Сергей получил в подарок необычную гирлянду. Эта Гирлянда состоит из N соединённых между собой лампочек. Все лампочки соединены в одну общую сеть в ней нет циклов. Непосредственно соединённые лампочки мы будем называть соседями.

В начальный момент, то есть, когда гирлянда подключена к сети ($t = 0$), некоторые лампочки включены, а некоторые выключены. Затем, каждую секунду, конфигурация включённых лампочек меняется. В момент времени t каждая лампочка либо включена, либо выключена в соответствии со следующим правилом:

- Лампочка не горит, если в момент $t - 1$ у нее была хотя бы одна соседняя лампочка, которая не горит;
- В противном случае лампочка горит.

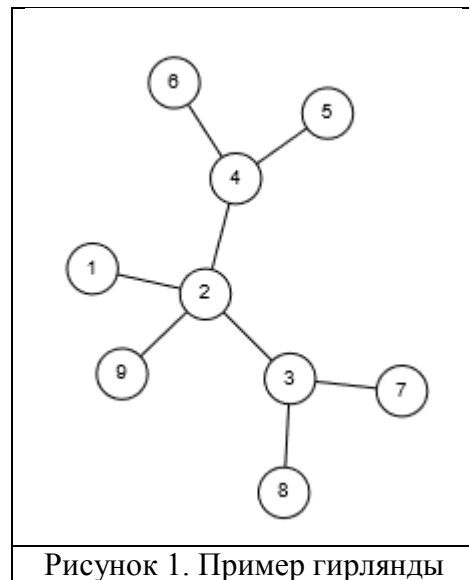


Рисунок 1. Пример гирлянды

Напишите программу, которая поможет Сергею узнать, через сколько секунд после включения в сеть все гирлянды погаснут.

Входные данные. Первая строка содержит одно целое число N —количество лампочек в гирлянде. Вторая строка содержит $N - 1$ целых чисел $p_i (2 \leq i \leq N, p_i < i)$. Число p_i означает, что i -я лампочка соединена с p_i -й лампочкой. Третья строка содержит N целых чисел $s_i (1 \leq i \leq N)$. Если $s_i = 1$, то лампочка i горит в нулевой момент времени, а если $s_i = 0$, то она не горит.

Результат должен содержать одно целое число - минимальное количество секунд до того, как все лампочки в гирлянде будут выключены. Если этого никогда не произойдет, выведите **-1**.

Исходные данные	Результаты	Пояснение
3 1 2 1 0 0	1	В нулевой момент времени горит только первая лампочка. В первый момент времени первая лампочка погаснет, а остальные - нет, поэтому ответ - 1 секунда.
3 1 2 1 0 1	-1	По крайней мере, одна лампочка всегда будет гореть.
9 1 2 2 4 4 3 3 2 1 0 0 0 1 1 1 1 1	1	Соответствует структуре на рисунке 1.

**Исходные данные:**

10
1 2 3 4 5 6 7 8 9
0 0 1 1 1 1 1 1 1 0

Решение:

Ключевое замечание заключается в том, что нам необходимо рассматривать четные моменты времени ($t = 0, 2, 4, \dots$) отдельно. Это связано с тем, что если лампочка не горит в какой-то момент времени, то через секунду она может снова загореться, но через две секунды она уже точно не будет гореть.

Поэтому для каждой лампочки l мы будем вычислять два параметра: $A[l]$, который обозначает первый нечетный момент времени, когда лампочка l не горит, и $B[l]$ - первый четный момент времени. Мы можем выразить эти параметры как расстояния от l до лампочек, которые не горят в начальный момент времени:

- $A[l]$ = кратчайшее расстояние нечетной длины от l до не горящей лампочки в начальный момент времени.
- $B[l]$ = кратчайшее расстояние четной длины от l до не горящей лампочки в начальный момент времени.

Для вычисления таких расстояний можно использовать различные стандартные графовые алгоритмы. Мы опишем метод, который особенно удобен, когда дерево представлено с указанием родительской вершины каждой вершины, как это сделано в данной задаче.

Лампа **1** является корнем дерева. Обратите внимание, что для перехода от вершины **u** к вершине **v**, мы пойдем одним из трех путей:

1. выйдя из **u**, мы всегда будем приближаться к корню, пока не достигнем **v**;
2. покидая **u**, мы удаляемся от корня, пока не достигнем **v**;
3. сначала мы приближаемся к корню, пока не достигнем общего родителя вершин **u** и **v**, а затем мы удаляемся от корня, пока не достигнем **v**.

Используя это наблюдение, расстояния $A[l]$ и $B[l]$ можно вычислить в четыре простых шага.

Шаг 1:

подготовка массива. Для каждой лампы l выберем начальные значения $A[l]$ и $B[l]$:

$$A[l] = \begin{cases} 0, & \text{если } l \text{ изначально не горит} \\ \infty, & \text{если } l \text{ изначально горит} \end{cases}$$

$$B[l] = \infty$$



Шаг 2:

двигаемся к корню. Мы рассматриваем вершины от последней до второй (*т.е.* $l = N, N - 1, \dots, 2$) и переносим расстояния на родительскую вершину:

$$A[\text{родитель}[l]] = \min\{A[l], B[l] + 1\},$$

$$B[\text{родитель}[l]] = \min\{B[l], A[l] + 1\}.$$

Шаг 3:

двигаемся от корня. Мы рассматриваем вершины из второго и последнего (*т.е.* $l = 2, 3, \dots, N$) и берем расстояния от родительской вершины:

$$A[l] = \min\{A[l], B[\text{родитель}[l]] + 1\},$$

$$B[l] = \min\{B[l], A[\text{родитель}[l]] + 1\}.$$

Шаг 4: поиск ответа. Первый момент, когда лампочка начинает гореть l , никогда больше не горит, равен $\max\{A[l], B[l]\} - 1$. Ответ можно получить, перебрав все лампочки l и выбрав максимальное значение. Если мы получим ответ ∞ , то это означает, что гирлянда никогда не погаснет, поэтому получаем **-1**

Решение на C++

```
#define _CRT_SECURE_NO_WARNINGS
```

```
#include<cstdio>
```

```
#include<algorithm>
```

```
using namespace std;
```

```
const int MAX_N = 500000;
```

```
const int INF = 1000000000;
```

```
// notLight[l][0] сохранит первый четный момент, когда лампочка l не горит
```

```
// notLight[l][1] сохранит первый нечетный момент, когда лампочка l горит
```

```
int notLight[MAX_N][2];
```

```
int parents[MAX_N];
```

```
int main() {
```

```
int N;
```

```
int result = 0;
```

```
scanf("%d", &N);
```

```
for (int i = 1; i < N; ++i) {
```

```
// переименуем вершины из 1,2,...i 0,1,...
```

```
scanf("%d", &parents[i]);
```

```
--parents[i];
```

```
}
```




```
for (inti = 0; i < N; ++i) {
    int d;
    scanf("%d", &d);

    // если лампочка не горит, установите notLight[i][0] = 0
    notLight[i][0] = (d ? INF : 0);
    notLight[i][1] = INF;
}

// движемся к родителю
for (inti = N - 1; i > 0; --i) {
    int t = parents[i];
    notLight[t][0] = min(notLight[t][0], notLight[i][1] + 1);
    notLight[t][1] = min(notLight[t][1], notLight[i][0] + 1);
}

// движемся от родителя
for (inti = 1; i < N; ++i) {
    int t = parents[i];
    notLight[i][0] = min(notLight[i][0], notLight[t][1] + 1);
    notLight[i][1] = min(notLight[i][1], notLight[t][0] + 1);
}

// мы находим лампочку
for (inti = 0; i < N; ++i) {
    int tm = max(notLight[i][0], notLight[i][1]) - 1;
    result = max(result, tm);
}

// Если ответ будет слишком большим, гирлянда не перестанет гореть.
if (result >= INF - 1)
    result = -1;

printf("%d\n", result);

return 0;
}
```

Решение на Python

```
MAX_N = 500000
INF = 10000000000

notLight = [[0, INF] for _ in range(MAX_N)]
parents = [0] * MAX_N

N = int(input())
result = 0
```



```
temp = list(map(int, input().split()))
for i in range(1, N):
    parents[i] = temp[i-1] - 1

temp2 = list(map(int, input().split()))
for i in range(N):
    d = temp2[i]
    notLight[i][0] = INF if d else 0
    notLight[i][1] = INF

for i in range(N - 1, 0, -1):
    t = parents[i]
    notLight[t][0] = min(notLight[t][0], notLight[i][1] + 1)
    notLight[t][1] = min(notLight[t][1], notLight[i][0] + 1)

for i in range(1, N):
    t = parents[i]
    notLight[i][0] = min(notLight[i][0], notLight[t][1] + 1)
    notLight[i][1] = min(notLight[i][1], notLight[t][0] + 1)

for i in range(N):
    tm = max(notLight[i][0], notLight[i][1]) - 1
    result = max(result, tm)

if result >= INF - 1:
    result = -1

print(result)
```

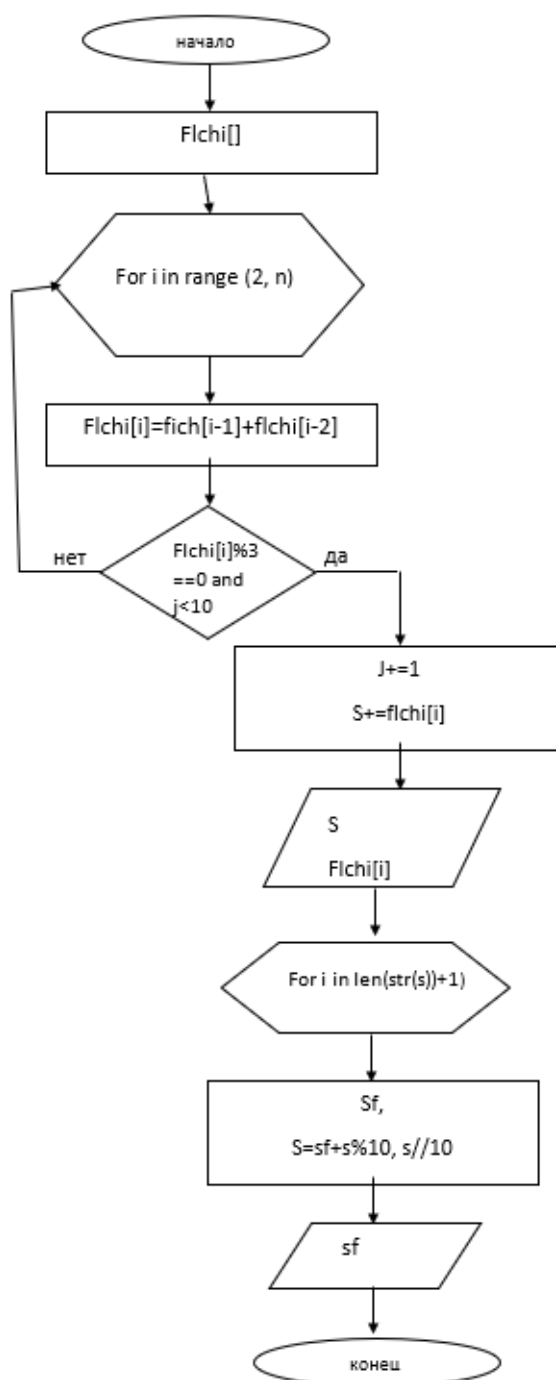
Задача 9.

Числа Фибоначчи – элементы числовой последовательности: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 610, 987, 1597, 2584, 4181, 6765, ..., в которой каждое последующее число равно сумме двух предыдущих чисел.

Среди чисел Фибоначчи встречаются числа, делящиеся на 3, 5 и 10. Определите сумму первых десяти чисел Фибоначчи, которые делятся на 3 и сумму цифр этих чисел.

Программа должна вывести первые 10 чисел Фибоначчи, делящиеся на 3, сумму этих чисел и сумму цифр этих чисел (не использовать встроенную или библиотечную функции вычисления чисел Фибоначчи).

Решение: в качестве решения сначала приведена блок-схема алгоритма:



Далее приведены примеры решения задачи на языке Python.

Решение с рекурсией (на Python):

```

def fichi(n):          # Рекурсивная функция вычисления числа Фибоначчи
    if n < 2:
        return 1
    else:
        return fichi(n-1)+fichi(n-2)
n, i, s, sf = 0, 0, 0, 0
print('Первые 10 чисел, делящиеся на 3:')
while True:           # организуем «вечный» цикл
    n += 1
  
```



```
f = fichi(n)      # вызов функции вычисления числа Фибоначчи
if f % 3==0:      # условие проверки делимости числа на 3
    i += 1
    print(i,' число =',f)  # вывод i-го числа Фибоначчи согласно условию
    s += f
if i >=10:        # условие выхода из «вечного» цикла: после вывода первых
10 чисел
    break         # Фибоначчи, делящихся на 3
print('Сумма первых 10 чисел: s=',s)
for i in range(len(str(s))+1):
    sf, s = sf + s%10, s//10
print('Сумма цифр в этой сумме: sf =',sf)
```

Решение без рекурсии (на Python):

j, s, sf, n = 0, 0, 0, 1000 # 1000 – максимальное количество чисел Фибоначчи,
задаем

```
fichi = [i for i in range(n)]
fichi[0], fichi[1] = 0, 1
print('Первые 10 чисел Фибоначчи, делящиеся на 3: ')
for i in range(2, n):
    fichi[i] = fichi[i-1] + fichi[i-2]
    if fichi[i]%3 == 0 and j<10:
        j +=1
        print(j,'число =',fichi[i])
        s += fichi[i]
print('Сумма первых 10 чисел: s =',s)
for i in range(len(str(s))+1):
    sf, s = sf + s%10, s//10
print('Сумма цифр в этой сумме: sf =',sf)
```

```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:29:22) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==== RESTART: D:\Олимпиада_шк_Гранит науки(2020в)\Фиб2020\fib_v3.py ====
Первые 10 чисел, делящиеся на 3:
1 число = 3
2 число = 21
3 число = 144
4 число = 987
5 число = 6765
6 число = 46368
7 число = 317811
8 число = 2178309
9 число = 14930352
10 число = 102334155
Сумма первых 10 чисел: s = 119814915
Сумма цифр в этой сумме: sf = 39
>>>
```

**Список рекомендуемой литературы для подготовки**

1. Босова Л.Л., Босова А.Ю., Информатика. 11 класс. Учебник. Базовый уровень, Просвещение, 2024 – 256с.
2. Семакин И.Г., Хеннер Е.К., Шеина Т.Ю., Информатика. Базовый уровень: учебник для 11 класса. – 3-е изд. – М.: БИНОМ. Лаборатория знаний, 2014. – 224 с.
3. К.Ю.Поляков, Е.А.Еремин, Информатика 11 кл. Учебник базового и углубленного уровней. В двух частях., 2019 – М.Бином – 302 с.
4. К.Ю.Поляков, Е.А.Еремин, , Информатика. 10-11 классы. Задачник. Базовый и углубленный уровни. Часть 1. ФГОС |, - Просвещение, 2023 – 240 с.
5. Лещинер В.Р., ЕГЭ 2024 Информатика: 16 вариантов. ОФЦ ТВЭЗ, Экзамен – 2024, - 183 с.
6. Анеликова Л. А., Лабораторные работы по Excel 3-е издание, - Салон-Пресс, 2022, - 112 с.
7. Белага В.В., Тятюшкина О.Ю., Основы системного анализа 10-11 классы. Учебник. ФГОС, 2019, 143с.
8. Долинский М.С. Решение сложных и олимпиадных задач по программированию. – СПб.: Питер, 2006. – 366 с.
9. Ерош И. Л., Сергеев М. Б., Соловьев Н. В. Дискретная математика СПб.: СПбГУАП, 2005. – 144 с.
10. Златопольский Д.М. Занимательная информатика.– М.: Бином.Лаборатория знаний, 2011. – 424 с.
11. Кирюхин В.М., Цветкова М.С. Информатика. Программы внеурочной деятельности учащихся по подготовке к Всероссийской олимпиаде школьников: 5–11 классы. – М.: БИНОМ. Лаборатория знаний, 2014. – 224 с.
12. Кудинов Ю. И., Пашенко Ф. Ф., Келина А. Ю. Практикум по основам современной информатики. – СПб. : Лань, 2011. – 352 с.
13. Малярчук С. Н. Информатика в определениях, таблицах и схемах. – Харьков: «Ранок». 2011 – 112с.
14. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы. 4-е изд. – СПб.: Питер, 2010. – 918 с.
15. Сафронов И.К. Готовимся к ЕГЭ. Информатика. – СПб.: BHV-Санкт-Петербург. 2009 – 368 с.
16. Калинин И.А., Искусственный интеллект. 10-11 классы. Учебное пособие. ФГОС, - Просвещение, 2023, - 144 с.
17. Триумфгородских М. Дискретная математика и математическая логика для информатиков, экономистов и менеджеров. – М.:«Диалог-МИФИ». 2011 – 180 с.

Составители: профессор каф. ИиКТ *Кризский В.Н.*, профессор каф. ИСВТ *Иванова И.В.*,
доцент каф. ИиКТ *Иванов И.В.*, доцент каф. ИиКТ *Кочнева А.А.*, ассистент каф. ИСВТ *Киба М.Р.*