XVII открытая заочная олимпиада

«Играем в программистов»

2021-22 учебного года

***«Реверси»***

Уважаемые участники олимпиады!

На этот раз все задачи объединены темой игра **«Реверси»**, **в самом распространённом её варианте** 8х8. Надеемся, что каждый из вас знает эту игру и её правила. Тем не менее, перед началом работы над задачами советую ознакомиться с ними (https://ru.wikipedia.org/wiki/Реверси).

Напомним, что игровое поле практически такое же, как и в шахматах, только поля не имеют цвета. Обозначение полей точно такое же, как и в шахматах, колонка обозначается заглавной латинской буквой, а строка – числом. Например, F3. После буквы пробела нет! Строки нумеруются снизу наверх.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |
|  | A | B | C | D | E | F | G | H |

***Уважаемые участники, с особой тщательностью прочтите нижеследующую информацию!***

***Система обозначений***

1. ***Текущее положение*** задаётся следующим образом: **во-первых**, задаётся расположение фишек на доске. **Во-вторых**, задаётся номер игрока, сделавшего последний ход (1 – ход первого игрока, 2 – второго), при этом в начальном положении задаётся 2.

|  |  |  |
| --- | --- | --- |
| Положение задаётся 8-ю строками. В каждой строке по 8 цифр (0, 1, 2). Каждая цифра соответствует одному полю.  На пустом месте находится НОЛЬ! 1 и 2 – фишки соответствующего игрока. | 00000000  00000000  00000000  00021000  00012000  00000000  00000000  00000000 | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000 |

1. ***Сделанный ход*** задаётся указанием поля, куда ставит свою фишку игрок. Например, E7.

***Входные данные*** читаются из файла **INPUT.txt**, а ***выходные данные*** записываются в файл **OUTPUT.txt** текущего каталога.

Внимательно читайте условия задач, к каждой из них приложены примерные тесты. Выбирайте задачи по своим силам.

Уважаемые участники! Предлагаемые задачи нельзя назвать лёгкими. **Если Вы не можете решить какую-либо задачу полностью, к чему надо стремиться, то решите её для посильных частных случаев!** Тогда за прохождение соответствующих тестов Вы получите некоторое количество баллов.

Все программы должны укладываться в 3 секунды.

Ещё один важный момент. В этой игре бывает, хотя и редко, что тот игрок, чья очередь ходить, не имеет такой возможности и вынужден пропускать ход, который переходит к противнику.

Напоминаем, что на нашей олимпиаде, как и на прочих, предполагается ввод ***корректных*** данных, то есть не противоречащих условию задачи. **Тратить время на «защиту от дурака» не надо!** Исключением являются лишь специальные задачи, в которых это сказано.

Не торопитесь! Задачи связаны друг с другом, и это может помочь в работе.

Желаем творческих успехов!

**Задача 1. Определение количества сделанных ходов (125 баллов).**

Задаётся расположение на доске. Сделать программу для определения количества сделанных ходов.

***Входные данные:*** 8 строк - положение.

***Выходные данные:*** одно число.

***Примерные тесты:***

|  |  |  |
| --- | --- | --- |
| ***Номер теста*** | **1** | **2** |
| ***Входные***  ***данные*** | 00000000  00000000  00000000  00021000  00012000  00000000  00000000  00000000 | 00000000  00000000  00010000  00011200  00011100  00000000  00000000  00000000 |
| ***Выходные данные*** | 0 | 3 |
| ***Комментарии*** | Начальное положение. | Должен быть сделан 4-й ход, ходит 2-й игрок. |

**Задача 2. Определение победителя и счёта (125 баллов).**

Задаётся окончательное расположение, то есть такое, в котором ни одна сторона не может сделать ход. Сделать программу, которая определит победителя и конечный счёт игры. Если победил 1-й игрок – вывести 1, 2-й игрок – 2, если ничья, то 0. Далее вывести количество баллов первого и второго игроков.

***Входные данные:*** 8 строк — окончательное положение.

***Выходные данные:*** строка с 3-мя числами через пробел: сперва число 1 / 2 / 0, в зависимости от результата игры, а затем количество набранных баллов первого и второго игроков.

***Примерные тесты:***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Номер теста*** | **1** | **2** | **3** |
| ***Входные***  ***данные*** | 12222222  21111112  22111222  21212112  22221222  22111122  22221212  22222121 | 22222221  11112221  11112221  11122221  11122221  11111111  11111111  11111111 | 12221222  21111112  22111222  21212112  22221222  22211111  22111111  22211111 |
| ***Выходные данные*** | 2 23 41 | 1 43 21 | 0 32 32 |

**Задача 3. Определение сделанного хода (125 баллов).**

Задаются два положения. Сделать программу, определяющую, можно ли из первого положения перейти ко второму за один ход, и, если возможно, выводящую этот ход. В случае невозможности, вывести «0». Возможен вариант, когда ход остаётся за предыдущим игроком.

***Входные данные:*** 18 строк — первое положение, второе положение.

***Выходные данные:*** обозначение хода, либо 0.

***Примерные тесты:***

|  |  |  |
| --- | --- | --- |
| ***Номер теста*** | **1** | **2** |
| ***Входные***  ***данные*** | 00000000  00000000  00000000  00021000  00012000  00000000  00000000  00000000  2  00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1 | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1  00000000  00000000  00000000  00021000  00012000  00000000  00000000  00000000  2 |
| ***Выходные данные*** | F4 | 0 |

**Задача 4. Определение получившегося положения (125 баллов).**

Задаётся положение и сделанный ход. Сделать программу, определяющую положение, которое получится в результате этого хода. Если указанный ход невозможен, выдать исходное положение. Конечное положение исключено.

***Входные данные:*** положение, очерёдность хода и ход.

***Выходные данные:*** положение.

***Примерные тесты:***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Номер теста*** | **1** | **2** | **3** |
| ***Входные данные*** | 00000000  00000000  00000000  00021000  00012000  00000000  00000000  00000000  2  F4 | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1  F3 | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1  G3 |
| ***Выходные данные*** | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1 | 00000000  00000000  00000000  00021000  00012100  00000200  00000000  00000000  2 | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1 |
|  |  |  | Ход невозможен, положение не меняется. |

**Задача 5. Составление списка возможных ходов (125 баллов).**

Задаётся текущее положение (расположение на доске и номер последнего сходившего игрока). Сделать программу, для составления списка всех возможных ходов того игрока, чья очередь ходить. Сперва вывести количество ходов, затем, сами ходы. Возможен случай, когда у очередного игрока нет хода, и ход не переходит. В случае окончательного положения выдать 0.

***Входные данные:*** положение.

***Выходные данные:*** три строки. В первой номер игрока, который должен сделать ход, во второй количество ходов, в третьей - ходы в отсортированном виде через пробел числа в одной строке через один пробел.

***Примерные тесты:***

|  |  |  |
| --- | --- | --- |
| ***Номер теста*** | **1** | **2** |
| ***Входные данные*** | 00000000  00000000  00000000  00021000  00012100  00000200  00000000  00000000  2 | 00000000  00000000  00000000  00111000  00012100  00000200  00000000  00000000  1 |
| ***Выходные данные*** | 1  4  C5 D6 E3 F2 | 2  5  C4 C6 E6 F5 G4 |

**Задача 6. Выигрыш в один ход (125 баллов).**

Задаётся неокончательное положение. Сделать программу, определяющую, может ли игрок, который должен ходить, выиграть в один ход. Если можно, выдать, номер игрока, кто побеждает и этот ход. В случае невозможности – выдать 0. Если выигрышных ходов больше одного, выдать все ходы в одной строке в упорядоченном вид.

***Входные данные:*** положение.

***Выходные данные:*** в зависимости от результата, либо 0, либо номер побеждающего игрока в первой строке и ход (ходы) во второй.

***Примерные тесты:***

|  |  |  |  |
| --- | --- | --- | --- |
| ***Номер теста*** | 1 | 2 | 3 |
| ***Входные данные*** | 00000000  00000000  00000000  00021000  00011100  00000000  00000000  00000000  1 | 00000000  00000000  00000100  00122220  00011111  00000100  00000100  00000000  2 | 12221012  21111112  22111222  21212112  22221222  22111122  22221212  22222121  2 |
| ***Выходные данные*** | 0 | 1  H5 | 2  F8 |
| ***Комментарии*** |  |  | Ход 2-го игрока (т.к. первый ходить не может!), он выигрывает ходом F8 |

**Задача 7. Наибольшее превращение за один ход (125 баллов).**

Задаётся неокончательное положение. Сделать программу, определяющую максимально возможное превращений фишек в результате очередного хода.

***Входные данные:*** положение.

***Выходные данные:*** три строки, в первой номер игрока, который должен ходить, во второй количество фишек, изменивших цвет, в третьей отсортированный список соответствующих ходов.

***Примерные тесты:***

|  |  |  |
| --- | --- | --- |
| ***Номер теста*** | **1** | **2** |
| ***Входные***  ***данные*** | 00000000  00000000  00000000  00021000  00012000  00000000  00000000  00000000  2 | 20000000  12100000  01200000  10221000  01121100  00022220  00010120  00000000  2 |
| ***Выходные данные*** | 1  1  C5 D6 E3 F4 | 1  5  D6 |
| ***Комментарии*** | Можно сделать лишь одно превращение. | Можно превратить 5 фишек. Жёлтым подсвечено место хода, голубым – места превращений. |

**Задача 8. Два хода подряд (125 баллов).**

Напоминаем, что в этой игре возможны такие положения (**неокончательные**), когда у одного из игроков после хода противника нет возможности ходить, и тогда ход остаётся за противником, и тот делает два хода подряд.

Сделать программу, определяющую может ли игрок, который должен ходить сделать два хода подряд. Если может, то выдать номер игрока и все такие ходы в отсортированном порядке. В случае невозможности выдать 0.

Если в заданном положении ход не переходит, то счёт ведётся заново.

***Входные данные:*** положение.

***Выходные данные:*** либо 0 (если невозможно сделать два хода подряд), либо 2 строки: в первой номер игрока, во второй – отсортированный список ходов.

***Примерные тесты:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Номер теста*** | 1 | 2 |  |  |
| ***Входные данные*** | 00000000  00000000  02000222  11111111  11111111  11111111  11111111  11111111  1 | 11111002  22222222  22111111  22221111  22221121  22222211  22222221  22222221  1 | 00000000  00000002  00000020  00111210  00012110  00011200  00210000  02000000  1 | 10000000  11000000  11120000  11112000  11111200  11111122  11111112  11111111  2 |
| ***Выходные данные*** | 1  A7 B7 C7 H7 | 0 | 0 | 1  F5 F6 G5 H4 |
| ***Комментарии*** |  | В начальном положении должен был бы ходить 2-й игрок, но у него хода нет. Значит, ходит 1-й игрок, но после его любого хода будет ходить 2-й игрок. |  |  |