Проекти за завършване на Java Sscript

курс 2016

# Крави и бикове

Игра, в която участникът трябва да познае намислено число. Играе се по следния начин:

* Компютърът намисля число
* Играчът въвежда число
* Компютърът извежда колко крави и бика има в числото
* Играчът въвежда ново число
* Компютърът извежда колко крави и бика има в новото число
* Играе се докато играчът не познае числото

Играчът има крава, ако във въведеното от него число има цифра, която я има в намисленото

Играчът има бик, ако във въведеното от него число има цифра, която я има в намисленото и тази цифра е на същата позиция в двете числа

Ограничения: намисеното число трябва да е четирицифрено, с различни цифри

Пример за игра:

* Намислено число 8456
* Въведено предположение 4231
* На екрана се изписва: 1 крава
* Въведено предполжение 4512
* На екрана се изписва: 2 крави
* Въведено предположение 5421
* На екрана се изписва: 1 крава, 1 бик
* и т.н.

# 2. Шифроване и дешифроване на Цезар шифър

Цезар шифър се нарича шифърът при който буквите от текст се изместват с определен брой напред.

Програмата ви трябва да може да шифрира и дешифрира цезар шифър.

Пример:

* Избирате шифриране
* Въвеждате текста за шифиране: Здравей!
* Въвеждате с какво отместване да се шифрира: 2 букви
* На екрана се изписва: Йжтвдзл!

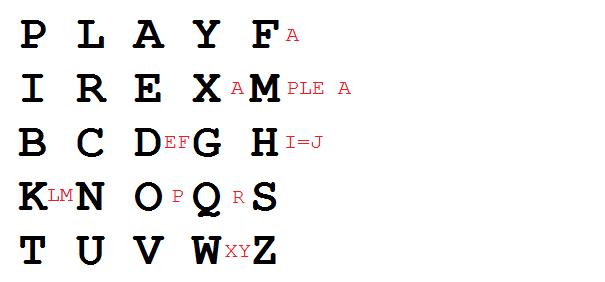
Пример 2:

* Избирате дешифриране
* Въвеждате текст за дешифриране: Йжтвдзл!
* Въвеждате с какво отместване е бил шифриран: 2 букви
* На екрана се изписва: Здравей!

# 3. Шифроване чрез Плейфеър шифър

<http://en.wikipedia.org/wiki/Playfair_cipher>

Метод за шифроване: думата-ключ се поставя в таблица 5х5, която се запълва с останалите букви от азбуката, така че да няма повтарящи се. Ако думата ключ е “playfair example”, то таблицата изглежда така:



Ако текста, който ще шифроваме е “Hide the gold in the tree stump” разделяме го на двойки букви. Преди това обаче добавяме X ако има еднакви букви една до друга.

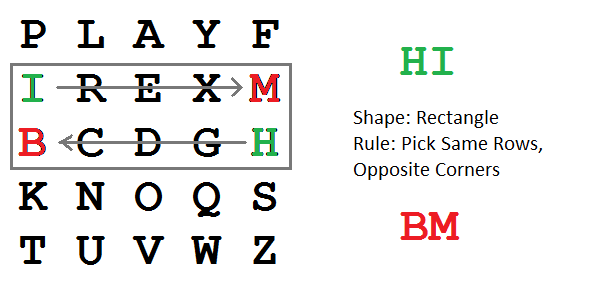
Резултата е:

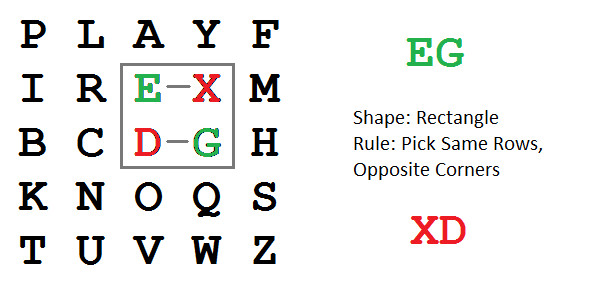
HI DE TH EG OL DI NT HE TR EX ES TU MP

Правилата на шифрирането са:

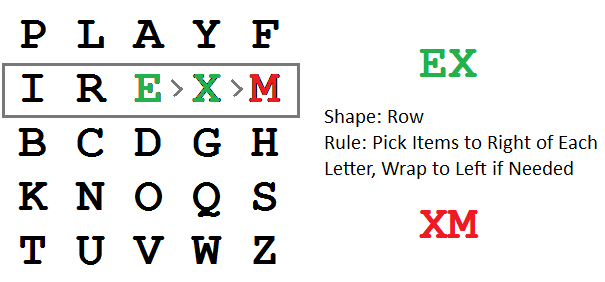
Намираме двойката букви в таблицата. Виждаме каква форма образуват.

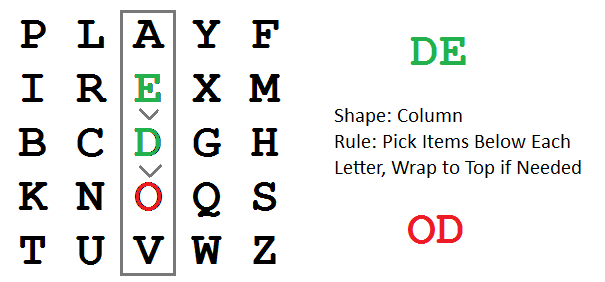
Ако образуват правоъгълник, за шифъра взимаме другите му два върха, като започваме от долния, ако в оригиналното двубуквие първа е долната, иначе вземаме горната..





Ако образуват колона или ред, вземаме следващата буква в колоната/реда.





Ако буквите са последни от дадена колона, взимаме първата буква на следващата колона. Същото за ред.

Пример:

* Въвежда се текста за шифроване
* Въвежда се думата-ключ
* Извежда се шифпования текст

# 4. Доклад

<http://judge.openfmi.net:9080/spoj0/descriptions.pl?problem_id=665>

# 5. Разбиване на Цезар шифър

При въведен криптиран текст да се изведе първоначалния, без да се задава какво е отместването.

Как става това:

* Знае се, че най-често срещаните букви в английския са e, t, h
* Намира се най-често срещания символ във въведения текст
* Пресмята се колко е отместването от него до e, t, h
* Извеждат се трите варианта за първоначален текст
* Ако потребителят не е доволен се извеждат останалите 23 варианта на текста

# 6.Candy crush

Функцията candyCrush1 връща true ако игралната дъска съдържа последователност от три или повече бонбона от един цвят. В противен случай - връща false.

Вашата задача е да проверите какъв ще бъде резултатът от функцията candyCrush1(newGameboard), след направен ход от дадена клетка - бонбон на игралната дъска. Позволените ходове са в рамките на игралната дъска - от позиция [0,0] не е възможно да се направи ход наляво и нагоре. В изходната позиция нямаме три последователни бонбона вертикално или хоризонтално в един цвят.

Ходовете са означени като L, R, U, D - ляво, дясно, горе, долу.

За примери и тестове, посетете линк към задачата в [CodeFights](https://codefights.com/challenge/eeABFLMYpFvSoe6Ls).

7. The Madman

Още веднъж ще трябва да се срещнете с изключителни сили, такива, които не сте очаквали. И сега ТОЙ идва, от страшна и луда действителност, която вие не можете да предположите, че същестува. ТОЙ идва да разруши времето и пространството!! (:))

За късмет, вие не само, че също сте луд, вие също сте и всесилен програмист. Знаете магическа рецепта, която ще защити вашия Свят от извънземната реалност, в която ТОЙ съществува, поне за известно време ….

Ето как работи рецептата:

* В нея влизат списък от съставки, които трябва да бъдат смесени във вашия магически казан една по една;
* Всяка съставка е различна, но много от тях са с едно и също магично действие /Досещате се - жабите и змийските зъби, например/
* Никога не трябва да добавяте две съставки с едно и също магично действие непосредствено една след друга /искате да предпазите Света, не да го разрушите, нали!/
* Всеки път, когато използвате рецептата си, предпазвате Света за един ден;
* Всеки път, когато следвате рецептата и смесвате продуктите в определена последователност, не можете да използвате тази последователност повече.

Като разполагате със списък от съставки за дадена рецепта и тяхното магическо действие, трябва да определите колко дни ще сте в състояние да опазите Света!

За примери и тестове, посетете линк към задачата в [CodeFights](https://codefights.com/challenge/Lyvgxy2D3kNoegvoX).

# 8. Пари

<http://judge.openfmi.net:9080/spoj0/descriptions.pl?problem_id=662>

# 9. Развитие

<http://judge.openfmi.net:9080/spoj0/descriptions.pl?problem_id=660>

# 10. Бойни кораби

Имате дъска 5х5. На нея компютърът е поставил 3 кораба. Всеки кораб заема само 1 квадратче. Вашата цел е да ги унищожите с най-малко ходове. Това става като на всеки ход въведете координатите където ще пуснете бомба. Ако не уцелите компютъра ви казва колко кораба има в реда и колоната на вашата атака.

<http://www.progressivejava.net/2012/10/Battleship-game-in-Java--How-to-program25.html>

11. caucusRace

Отправена ви е покана за следната игра. Имате цели числа, написани едно след друго в нещо като кръг /в случая точната форма, не е от значение/. Играчът започва играта със сметка 0 и избира позиция от кръга i.

Числото values[i], написано в тази позиция се добавя към сметката на играча. Всеки ход, ход който прави играчът са по посока на часовниковата стрелка. При всеки ход, сметката на играча се променя със стойността на числото изписано на позицията, на която стъпва. Играта свършва и играчът губи, ако след направен ход баланса му стане равен или по-малък от нула. Играчът печели, ако успее да премине целия кръг.

Вашата задача е - като получите списък със стойностите на кръга, да върнете списък с печелившите начални позиции.

За примери и тестове, посетете линк към задачата в [CodeFights](https://codefights.com/challenge/XRMzBvaNTmSkSKZjt).

12. Змия

Разполагате с правоъгълна дъска, състояща се от n × m клетки. Върху някои от клетките е разположена змия, другите клетки са свободни. Змията се състои от две или повече последователни клетки, допиращи се вертикално или хоризонтално. Първата клетка е главата на змията, последната е нейната опашка.

При всеки ход, главата на змията се придвижва към една от допиращите се до нея вертикално или хоризонтално клетки, при това клетките, част от змията, се изместват в същата посока. Преместването на клетките става едновременно и е възможно главата да заеме мястото на опашката. Забранени са две конфигурации на клетките, които са част от змията - главата да пресече тялото на змията и излизане извън границите на дъската.

Пътят на змията е поредица от знаците L, R, D, и U (съответно ляво, дясно, надолу и нагоре), описващи движението на главата на змията при всеки ход.

Вашата задача е да определите колко различни пътя може да измине змията по дъската?

За примери и тестове, посетете линк към задачата в [CodeFights.](https://codefights.com/challenge/dPB9sdu3E5HHRjZeY)