# Изпит по "Основи на програмирането"

## Задача 6. Смени

Любимият отбор на Пепи е на финал, но започва да губи мача. Треньорът на отбора не знае какви смени да направи, за да обърне резултата. Напишете програма, с която ще разберете кой са **първите 6 смени**, които могат да се направят. Знаем, че всяка цифра от двата номера е в даден интервал. За да бъде **възможна една смяна**, **първата цифра от номера трябва да бъде четна, а втората - нечетна**.

### Вход:

От конзолата се четат 4 реда:

* **K** - началото на интервала за първото число от първия номер – **цяло число в интервала [0..8]**
* **L** - началото на интервала за второто число от първия номер – **цяло число в интервала [9..0]**
* **M** - началото на интервала за първото число от втория номер – **цяло число в интервала [0..8]**
* **N** - началото на интервала за второто число от втория номер – **цяло число в интервала [9..0]**

### Изход:

На конзолата да се отпечатат **първите 6 възможни смени** по следния начин:

* Ако смяната **е възможна и номерата НЕ съвпадат,** да се отпечата:

**"{K}{L} - {M}{N}"**

* Ако смяната **е възможна и номерата съвпадат**, да се отпечата:

**"Cannot change the same player."**

### Примерен вход и изход:

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| 7  6  8  5 | Cannot change the same player.  89 - 87  89 - 85  87 - 89  Cannot change the same player.  87 - 85 | За първия номер знаем, че: Първата цифра е в интервала [7;8], втората цифра е в интервала [9;6].  Аналогично, за втория номер: Първата цифра е в интервала [8;8], втората цифра е в интервала [9;5].  Започваме да генерираме всички възможни комбинации от цифри в тези интервали. Първата генерирана комбинация е 79 – 89. Тя обаче **не е валидна**, понеже не изпълнява условието **първата цифра на всеки номер да е четна, а втората нечетна**. Следващата комбинация е 79 – 88. Тя също не изпълнява условието, следователно **не я принтираме,** и т.н.  Стигаме до комбинацията 89 – 89, която е **валидна**,но понеже **двата номера са еднакви**, принтираме съответния изход. Следващата комбинация е 89 - 88, която е невалидна. След нея е 89 – 87, която е валидна и я принтираме. Така **броят на смените става 1**. Продължаваме да генерираме комбинации и стигаме до 87 – 85, която е четвъртата валидна смяна. В конкретния случай нямаме повече валидни смени, за това не са принтирани 6 такива. |
| 6  7  5  6 | Cannot change the same player.  69 - 67  69 - 89  69 - 87  67 - 69  Cannot change the same player.  67 - 89  67 - 87 | Започваме да генерираме комбинациите от дадените числа. Първата комбинация е 69-59, но е **невалидна и не я принтираме**. Втората е 69 – 58, но също е невалидна. Продължаваме и стигаме до комбинацията 69-69, която е **валидна**, но понеже **номерата са еднакви** принтираме: Cannot change the same player.  Следващата комбинация 69-68 е невалидна. Следва 69-67, която е валидна и я принтираме, така **броят на смените става 1**, и т.н.  Стигаме до валидната комбинация 67-87. Принтираме я и **броят на смените става 6**. Програмата **приключва**. |

### JavaScript - Примерен вход и изход:

|  |  |  |
| --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** |
| (["7",  "6",  "8",  "5"]) | Cannot change the same player.  89 - 87  89 - 85  87 - 89  Cannot change the same player.  87 - 85 | За първия номер знаем, че: Първата цифра е в интервала [7;8], втората цифра е в интервала [9;6].  Аналогично, за втория номер: Първата цифра е в интервала [8;8], втората цифра е в интервала [9;5].  Започваме да генерираме всички възможни комбинации от цифри в тези интервали. Първата генерирана комбинация е 79 – 89. Тя обаче **не е валидна**, понеже не изпълнява условието **първата цифра на всеки номер да е четна, а втората нечетна**. Следващата комбинация е 79 – 88. Тя също не изпълнява условието, следователно **не я принтираме,** и т.н.  Стигаме до комбинацията 89 – 89, която е **валидна**,но понеже **двата номера са еднакви**, принтираме съответния изход. Следващата комбинация е 89 - 88, която е невалидна. След нея е 89 – 87, която е валидна и я принтираме. Така **броят на смените става 1**. Продължаваме да генерираме комбинации и стигаме до 87 – 85, която е четвъртата валидна смяна. В конкретния случай нямаме повече валидни смени, за това не са принтирани 6 такива. |
| (["6",  "7",  "5",  "6"]) | Cannot change the same player.  69 - 67  69 - 89  69 - 87  67 - 69  Cannot change the same player.  67 - 89  67 - 87 | Започваме да генерираме комбинациите от дадените числа. Първата комбинация е 69-59, но е **невалидна и не я принтираме**. Втората е 69 – 58, но също е невалидна. Продължаваме и стигаме до комбинацията 69-69, която е **валидна**, но понеже **номерата са еднакви** принтираме: Cannot change the same player.  Следващата комбинация 69-68 е невалидна. Следва 69-67, която е валидна и я принтираме, така **броят на смените става 1**, и т.н.  Стигаме до валидната комбинация 67-87. Принтираме я и **броят на смените става 6**. Програмата **приключва**. |