O candidato terá 4 horas para resolver os exercícios abaixo. As implementações podem ser realizadas em qualquer linguagem. Pode-se resolver os exercícios em qualquer ordem. Quando encerrar o tempo ou as atividades, o código deve ser enviado à um repositório no github ou enviado por e-mail.

### 

### Travessia do rio

Em uma margem de um rio está um camponês com um leão, uma cabra e um fardo de feno. Ele precisa atravessar o rio da margem esquerda para margem direita, contudo seu barco consegue levar somente o camponês e 1 dos 3 objetos. O problema é que caso o camponês não esteja presente na margem, o leão irá comer a cabra e a cabra irá comer o feno, ou seja, o algoritmo deve buscar soluções onde todos cheguem até a outra margem. - Importante que não deve-se implementar uma das soluções (movimentar cabra, retornar sozinho, movimentar leão, retornar com cabra, movimentar feno, retornar sozinho e movimentar cabra novamente), mas sim que o algoritmo saiba chegar à uma solução.

Ao executar o programa, ele deve executar em algum output (console) as movimentações que foram feitas para chegar até o resultado.

Por exemplo

-> camponês e cabra para margem direita

-> camponês para margem esquerda

-> camponês e leão para margem direita

-> camponês e cabra para margem esquerda

-> camponês e feno para margem direita

-> camponês para margem esquerda

-> camponês e cabra para margem direita

### Número romanos

Implemente um programa que recebe como entrada um número entre 1 e 1000 e retorne o número no formato romano. Números disponíveis em <https://www.somatematica.com.br/fundam/tabela1.php>

Novamente, basta escrever o resultado em qualquer output (console)

Entrada: 858

Saída: DCCCLVIII

### União e intersecção de conjuntos

Implemente um programa que irá receber 2 sequências de números como entrada (pode ser no console, separado por vírgula) e então irá escrever a união e intersecção desses conjuntos.

Por exemplo,

Primeiro conjunto: 1,2,6,7

Segundo conjunto: 1,3,8

--

Intersecção: 1

União: 1,2,3,6,7,8