

# Trabalho Prático 1

---

## Qual é o problema?

Travian é um jogo on-line MMO de estratégia muito conhecido no mundo todo. Ao iniciar o jogo, o jogador vira o chefe de uma aldeia, podendo escolher ser de um de cinco povos - Gauleses, Romanos, Teutões, Egípcios, Hunos e Espartanos. O jogador controla uma região envolta da aldeia da qual os aldeões retiram os recursos, que são madeira, barro, ferro e cereal. O jogador pode escolher pagar para aumentar o nível das regiões, ao custo de uma certa quantidade de recursos, assim produzindo mais destes naquele local. Na aldeia o jogador pode criar novas construções e evoluir construções existentes, deixando-as mais eficientes. O jogador pode ainda construir uma muralha para se defender de possíveis inimigos.

Atualmente existem servidores em todos os continentes, e até servidores intercontinentais, onde cada jogador deve jogar uma eliminatória nacional para ai sim se qualificar para um servidor internacional que conta com mais de 50 nacionalidades, aumentando assim o nível dos jogadores e a competição. E estes servidores podem funcionar em diferentes velocidades de tempo. Por exemplo 3x(o tempo no mundo virtual passa três vezes mais rápido que no mundo real).

Mais informações sobre o jogo podem ser encontradas em

- <https://support.travian.com/en/support/home> e
- <https://www.travian.com/br>
- [https://travian.fandom.com/wiki/Travian\\_Wiki](https://travian.fandom.com/wiki/Travian_Wiki)

Em todos os servidores, independente da sua velocidade no mundo virtual, existe uma período inicial em que o jogador não pode ser atacado. Normalmente esse período é de 72h no mundo real. Sendo assim, nós desejamos saber como aproveitar da melhor maneira possível estas 72 horas iniciais do jogo para tomar a frente no servidor.

Dessa forma vamos desenvolver um modelo matemático para maximizar o número de habitantes em nossa aldeia no intervalo de planejamento de 72h iniciais. Observe que existem regras limitando a construção e/ou melhora de alguns prédios e também dos campos de recursos. Estas regras devem ser respeitadas para que o plano proposto seja viável. No jogo existe a possibilidade de se comprar uma moeda virtual que permite fazer melhorias automaticamente, para nosso trabalho esta característica será desconsiderada, assim como não consideraremos a existência do herói e dos pontos/recursos que são conseguidos com ele. Entretanto, cada um dos povos tem características e prédios únicos que afetam como o jogo é realizado. Por exemplo, os romanos podem evoluir ao mesmo 3 edifícios enquanto os demais povos só podem construir 2, entretanto seus prédios são mais caros e mais lentos para construção.

Outro fator importante é que desejamos que ao final dessas 72h iniciais desejamos ter uma força de pelo menos 100 soldados básicos de infantaria (dependente do povo escolhido).

Assim, os alunos se dividiram em equipes e receberão uma tribo e uma velocidade de mundo do servidor. Eles devem então desenvolver um modelo de programação matemática que maximize o tamanho da população daquela tribo naquele servidor ao final de 72h com pelo menos a criação de 100 soldados de infantaria básicos.

## Algumas Suposições

Algumas suposições/explicações:

- Só é necessário modelar as evoluções de prédios e recursos. Outros recursos como segunda vila, embaixador, herói e etc são irrelevantes para o trabalho.
- Assuma que os novos recursos chegam de **hora em hora**. Ou seja, os recursos só aumentam ao final de 1 hora no mundo do jogo. Observe que eles ainda devem respeitar o limite de capacidade do armazém e celeiro.
- Você só pode fazer uma construção de cada tipo, assim só pode haver um esconderijo, um armazém e etc.
- O jogo inicia apenas com o edifício principal no nível 1, um campo de cada recurso no nível 1 e 750 de cada recurso.
- Ignore o bônus de aceleração dada pela construção principal. Considere que os tempos da construção principal no nível 1.
- Para produzir os soldados, considere que o prédio escolhido para a produção é o (**quartel**) que precisa ter suas dependências satisfeitas.
- Considere que os romanos podem construir 3 *coisas* simultaneamente, independente de serem recursos ou prédios.
- Sobre os romanos, considere que o tempo de construção e preço pode assumir que são 30% maiores que os padrão.
- Assuma que as tropas são treinadas de forma imediata. Ou seja, uma vez os recursos necessários alcançados, pode-se fazer as tropas imediatamente e com não há tempo de treinamento.

## Como é a equipe e escolha do povo?

Os alunos poderão formar grupos de até 3 componentes e devem enviar um email para o email [marciocs@dcc.ufmg.br](mailto:marciocs@dcc.ufmg.br) com o nome completo e matrícula de todos os integrantes. Os alunos receberão como resposta um email contendo uma ID para identificação da equipe e a tribo e velocidade do servidor com o qual vão trabalhar.

## Como será a entrega?

A entrega será realizada pelo Moodle em tarefa correspondente. Existirão duas tarefas, uma para o código e uma para o relatório e apenas um dos alunos da equipe deve fazer o upload da solução de toda a equipe e deve informar o ID da equipe no campo de comentários da submissão.

## O que deve ser entregue?

A equipe deve entrar dois documentos: um código em julia com o modelo implementado e um relatório PDF com a explicação do modelo e a descrição da solução final obtida pelo modelo. A equipe precisa descrever as escolhas de modelagem e as suposições feitas no documento de forma clara e o programa deve poder ser executado por linha de comando com o comando: `julia <nomeprograma>`

## Como será a avaliação?

A tarefa vale 10 pontos e estes serão divididos igualmente entre o relatório e o código. Ou seja, o relatório vale 5 pontos e o código vale 5 pontos. Na avaliação do relatório será considerada a clareza e redação da solução do problema e na avaliação do código será considerada apenas se o mesmo funciona e está correto.

## Pontuação EXTRA

As equipes que desejarem uma pontuação extra, poderão adquirir até 5 pontos ao integrar algumas características particulares do jogo que podem tornar a solução mais prática. A seguir descrevemos cada uma delas e a pontuação extra provida por cada uma.

**Uso de ouro (1 pt extra):** assuma que temos 100 golds (moeda) do jogo no início do jogo. Essas moedas são usadas para melhorar automaticamente uma construção. Como podemos incorporar essa característica na nossa solução?

**Tempo de Jogo (4 pt extra):** as soluções obtidas aqui, apesar de ótimas, não são práticas pois (normalmente) ninguém pode jogar 24 horas por dia. Dessa forma, assuma que temos uma nova entrada para o modelo: as janelas de tempo que o jogador está disposto a jogar. Assim, o seu modelo será modificado de acordo com a janela de tempo escolhida pelo jogador dentro das 72 horas. O seu código deve receber um arquivo de texto com os intervalos de jogo do jogador nos três dias. Os intervalos serão dados com horas e meias horas e no início do arquivo estará a quantidade de intervalos de tempos que o jogador vai jogar. Por exemplo

```
3
1:00 7:30
9:00 10:00
25:00 27:30
```

Significa que o jogador tem 3 horários disponíveis. Da hora 1:00 do início do servidor até a hora 7:30; Da hora 9:00 até as 10:00 e da hora 25:00 até 27:30.