

Usando Gurobi como solver do JuMP em Julia

1. Instalação

- Clique em <https://pages.gurobi.com/registration> para fazer uma conta
- Selecione a opção `Academic`
- Preencha com seus dados pessoais; Em `Company Email Address`, coloque o e-mail de sua universidade e em `University` o nome.
- Em `Academic Position` selecione a opção `Student`
- Por fim, clique em `Access Now` e confirme o registro em seu e-mail
- Clique em <https://www.gurobi.com/login/> e logue no site com sua conta
- Clique em <https://www.gurobi.com/downloads/gurobi-optimizer-eula/> e aperte em `I Accept the End User License Agreement`.
- Se seu sistema operacional for Windows, clique em `Gurobi-9.0.2-win64.msi`. Se for Linux, clique em `gurobi9.0.2_linux64.tar.gz` e se for macOS, `gurobi9.0.2_mac64.pkg`.
- Após o término, siga os procedimentos do <https://packages.gurobi.com/9.0/README.txt> para instalação

2. Licença Acadêmica

- Clique em <https://www.gurobi.com/downloads/end-user-license-agreement-academic/> e selecione a opção `I Accept These Conditions`
- Vá mais abaixo na página até a parte de `Installation`. Copie o código que começa com `grbgetkey ...`.
- No Windows, abra o `Command Prompt` ao digitar em pesquisa por `cmd` e cole esse código lá. No Linux e no macOS, cole o código diretamente no terminal
- Siga os procedimentos e aceite o diretório de instalação sugerido

3. Configuração no Atom

- Abra o atom e inicialize o Julia. Adicione o JuMP digitando no REPL `Pkg.add("JuMP")`
- Adicione o pacote do Gurobi digitando no REPL `Pkg.add("Gurobi")`
- Em seguida, é necessário fazer o build. Digite no REPL `Pkg.build("Gurobi")`
- Após esses passos basta chamar as bibliotecas utilizando o `using`