

# Projeto 01

Aluno: Luiz André da Silva Carvalho - 11911BSI225

Repositório: [https://github.com/oluizcarvalho/genetic\\_algorithm\\_knapsack](https://github.com/oluizcarvalho/genetic_algorithm_knapsack)

## Sobre o projeto:

Cada algoritmo tem a função `main` que irá resolver o problema da mochila para os 4 primeiros arquivos de entrada da pasta *input*, os resultados são mostrados através de logs no terminal e também é gerado ou atualizado um arquivo na pasta *output*.

`/ag_chromosome.py` → Algoritmo genético de cromossomos

`/dynamic.py` → Algoritmo de programação dinâmica

`/grasp.py` → Algoritmo meta-heurística GRASP

`/report.py` → Algoritmo que roda os outros algoritmos, gera gráficos com a média e desvio padrão dos resultados e tempo de execução

## Váriaveis

- Algoritmo genético de cromossomos
  - `probabilidade_mutacao = 0.2` *Probabilidade de mutações*
  - `geracoes = 100` *Número de gerações*
- GRASP
  - `alpha = 0.3` *Alpha*
  - `max_iterations = 100` *Máximo de iterações*
- Reports
  - `num_execucoes = 15` *Número de vezes que cada algoritmo será executado*
  - `num_arquivos = 4` *Número de instâncias rodadas pelos outros algoritmos*

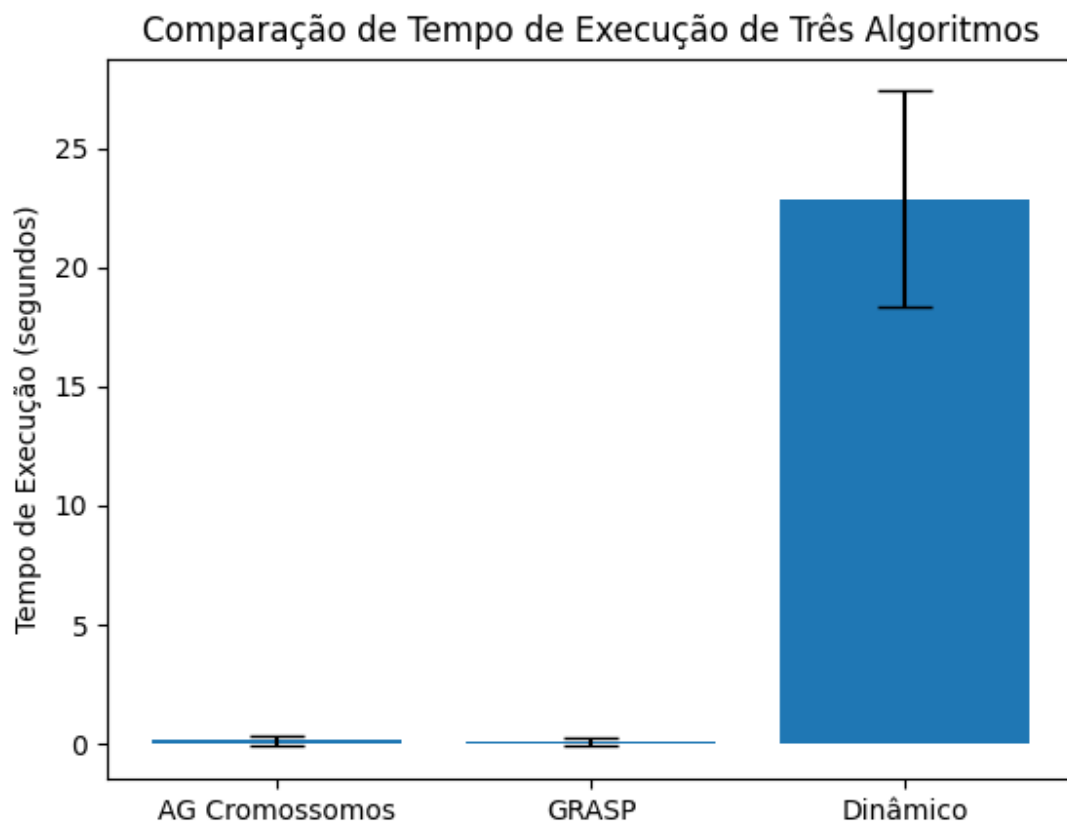
P.s:

- O algoritmo de programação dinâmica e GRASP forem baseados desse repositório.

## Relatório

### Comparação de performance

O gráfico foi gerado com base na execução de 15 vezes cada algoritmo



## Média

Algoritmo	Métrica
AG	0.11374131838480632
GRASP	0.023114856084187826
Dinâmico	22.385113938649496

## Desvio padrão

Algoritmo	Métrica
AG	0.006742787448252973
GRASP	0.002266955136276829
Dinâmico	0.2770282241543022

### 1. Médias:

- O algoritmo "GRASP" tem a menor média de tempo de execução entre os três algoritmos, indicando que ele é o mais rápido em média.
- O algoritmo "AG" tem uma média intermediária, com tempos de execução mais longos do que o "GRASP" e mais curtos do que o "Dinâmico".
- O algoritmo "Dinâmico" tem a maior média de tempo de execução, sugerindo que ele é o mais lento em média.

### 2. Variação:

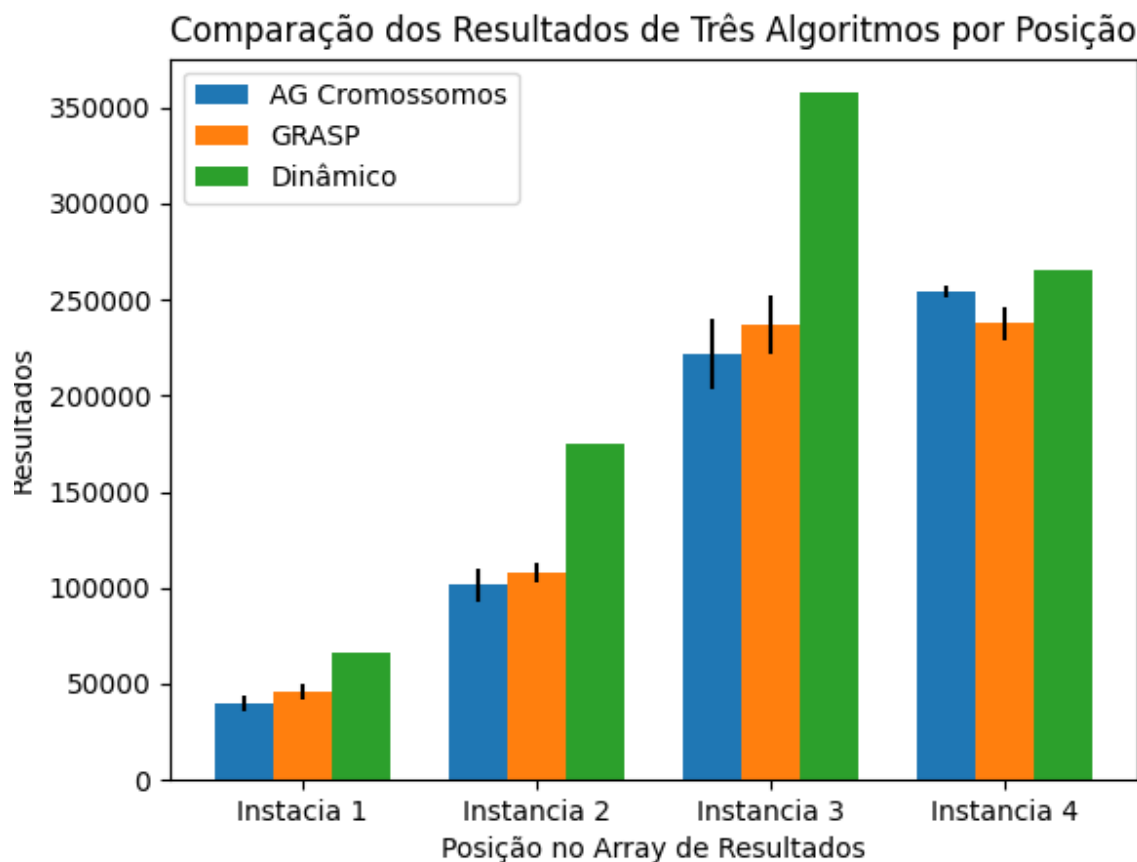
- Os desvios padrão relativamente baixos para todos os algoritmos indicam que os tempos de execução tendem a ser consistentes e não variam muito em relação à métrica de tempo.

### 3. Comparação Geral:

- Se a métrica for o tempo de execução e o objetivo for a execução mais rápida, o "GRASP" é a escolha preferencial, pois tem a menor média de tempo de execução.
- O "AG" oferece tempos de execução intermediários em relação aos outros dois algoritmos.
- O "Dinâmico" é o mais lento em média.

## Comparação de resultados

O gráfico foi gerado com base na execução de 15 vezes cada algoritmo, menos o dinâmico que não sofre alteração a cada execução.



### Média por algoritmo e instância:

Algoritmo	Instância 1	Instância 2	Instância 3	Instância 4
AG	41492.4	100079.4	238371.4	255314.33
GRASP	43130.2	110113.4	238620.2	241986.33
Dinâmico	66391.0	175269.0	357639.0	265254.0

### Desvio padrão:

Algoritmo	Instância 1	Instância 2	Instância 3	Instância 4
AG	3606.04	7087.82	17297.69	3289.55
GRASP	3102.73	7573.67	10051.34	4672.16
Dinâmico	0.0	0.0	0.0	0.0

### 1. Valores Médios:

- O algoritmo "Dinâmico" geralmente retorna os maiores valores médios em todas as instâncias.
- O algoritmo "GRASP" também tem valores médios relativamente altos, embora ligeiramente menores do que o "Dinâmico".
- O algoritmo "AG" retorna os valores médios mais baixos entre os três algoritmos.

### 2. Variação de Valores:

- O algoritmo "AG" apresenta uma variação significativa nos valores retornados entre as instâncias, conforme evidenciado pelos desvios padrão relativamente altos.
- O algoritmo "GRASP" tem uma variação intermediária, com desvios padrão menores em comparação com o "AG".
- O algoritmo "Dinâmico" retorna valores muito consistentes, pois seus desvios padrão são zero em todas as instâncias.

### 3. Consistência do Algoritmo Dinâmico:

- O algoritmo "Dinâmico" demonstra alta consistência em seus resultados, produzindo valores consistentemente altos.

## Resumo

Em relação aos resultados apresentados podemos ver que o algoritmo dinâmico, como esperado apresenta o melhor resultado, porém a custo de muito tempo de execução, ficando a critério de quem está executando avaliar se o custo compensa.

Os algoritmos GRASP e AG apresentaram resultados semelhantes, se mostraram muito performáticos com as variáveis escolhidas, GRASP apresentou uma performance um pouco melhor e também não ficou atrás nos resultados e com um desvio padrão menor que o AG, então pode se supor que com as variáveis escolhidas o GRASP demonstrou o melhor custo benefício levando em consideração consistência dos dados, resultados e tempo de execução.