

Relatório 7 - Bruna Meinberg

1. Busca exaustiva

Utilizando a busca exaustiva para rodar as novas entradas, obtive os seguintes resultados:

	tempo em segundos	melhor peso (de acordo com o melhor valor)	melhor valor
entrada 1	4,57166	373	601
entrada 2	97,9007	239	870
entrada 3	não sei ;(não sei ;(não sei ;(
entrada 4	12,4119	600	648

2. Tarefa 1

Para a tarefa 1, o código de busca foi executado 5x para cada entrada, obtendo os seguintes resultados:

	tempo de execução	melhor peso (de acordo com valor)	melhor valor
entrada 1	0,00147765	372	369
entrada 2	0,00188213	237	350
entrada 3	0,00173881	405	3320
entrada 4	0,00161607	594	526

3. Tarefa 2

Para a tarefa 2, o código de busca foi executado 5x para cada entrada, obtendo os seguintes resultados:

	tempo de execução	melhor peso (de acordo com valor)	melhor valor
entrada 1	0,00140732	378	343
entrada 2	0,00175318	236	312
entrada 3	0,00176369	259	1888

entrada 4	0,00164743	560	447
-----------	------------	-----	-----

4. Comparando as tarefas

No geral, o código construído com para a tarefa 1 teve um maior desempenho qualitativo, obtendo valores maiores. Agora em relação ao tempo, praticamente não houve diferença. Então, para analisar os tempos, vou aumentar de 5 vezes para 100.000 vezes com o intuito de checar se há grande divergência de tempo. Para o teste, vou usar somente a entrada 1:

	Tempo
Tarefa 1	0,501624
Tarefa 2	0,588408

Por mais que o tempo da tarefa 1 tenha sido melhor, não houve uma grande diferença. Tentando com 1.000.000

	Tempo
Tarefa 1	5,42066
Tarefa 2	6,47776

Visivelmente, com o aumento do número, a tarefa 1 toma menos tempo. Em ambos os casos, a tarefa 1 teve um melhor desempenho também na determinação do valor/peso.

4. Comparando as tarefas com a busca exaustiva

Por mais que os tempos da busca exaustiva sejam maiores, eles podem ser benéficos para seu objetivo. Se o que você busca é precisão, é interessante utilizar um algoritmo como a busca exaustiva, para ter a “certeza” de que sua resposta é a melhor resposta. Agora, se não necessariamente você precisa da melhor resposta possível, pode ser interessante utilizar as construções aleatórias, que tem um bom resultado (a depender do número de iterações) e demora muito menos tempo.