nhança sempre será mínimo local para qualquer outra vizinhança. Para um problema de minimização, uma solução ótima é mínimo local para qualquer vizinhança. Heurísticas são sempre não determinísticas, pois não tem garantia de qualidade da solução. Em uma heurística de busca sempre é mais eficiente computarmos a solução calculando a diferença do valor em relação a solução anterior. O algoritmo Multi-Start converge eventualmente para uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande. (3 pontos) Dada uma instância problema do Caixeiro Viajante		Nome:
nhança sempre será mínimo local para qualquer outra vizinhança. Para um problema de minimização, uma solução ótima é mínimo local para qualquer vizinhança. Heurísticas são sempre não determinísticas, pois não tem garantia de qualidade da solução. Em uma heurística de busca sempre é mais eficiente computarmos a solução calculando a diferença do valor em relação a solução anterior. O algoritmo Multi-Start converge eventualmente para uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande. (3 pontos) Dada uma instância problema do Caixeiro Viajante com 10 cidades quantos vizinhos uma solução tem na vizinhança 20PT?	1.	(5 pontos) Verdadeiro ou falso?
é mínimo local para qualquer vizinhança. Heurísticas são sempre não determinísticas, pois não tem garantia de qualidade da solução. Em uma heurística de busca sempre é mais eficiente computarmos a solução calculando a diferença do valor em relação a solução anterior. O algoritmo Multi-Start converge eventualmente para uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande. 2. (3 pontos) Dada uma instância problema do Caixeiro Viajante com 10 cidades quantos vizinhos uma solução tem na vizinhança 2OPT? 3. (2 pontos) Compare o algoritmo de Metropolis com uma Têm-		Uma solução que é mínimo local para uma dada vizi- nhança sempre será mínimo local para qualquer outra vizinhança.
tem garantia de qualidade da solução. Em uma heurística de busca sempre é mais eficiente computarmos a solução calculando a diferença do valor em relação a solução anterior. O algoritmo Multi-Start converge eventualmente para uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande. 2. (3 pontos) Dada uma instância problema do Caixeiro Viajante com 10 cidades quantos vizinhos uma solução tem na vizinhança 2OPT? 3. (2 pontos) Compare o algoritmo de Metropolis com uma Têm-		 Para um problema de minimização, uma solução ótima é mínimo local para qualquer vizinhança.
computarmos a solução calculando a diferença do valor em relação a solução anterior. O algoritmo Multi-Start converge eventualmente para uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande. 2. (3 pontos) Dada uma instância problema do Caixeiro Viajante com 10 cidades quantos vizinhos uma solução tem na vizinhança 2OPT? 3. (2 pontos) Compare o algoritmo de Metropolis com uma Têm-		 Heurísticas são sempre não determinísticas, pois não tem garantia de qualidade da solução.
uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande. 2. (3 pontos) Dada uma instância problema do Caixeiro Viajante com 10 cidades quantos vizinhos uma solução tem na vizinhança 2OPT? 3. (2 pontos) Compare o algoritmo de Metropolis com uma Têm-		Em uma heurística de busca sempre é mais eficiente computarmos a solução calculando a diferença do valor em relação a solução anterior.
com 10 cidades quantos vizinhos uma solução tem na vizinhança 2OPT? 3. (2 pontos) Compare o algoritmo de Metropolis com uma Têm-		O algoritmo Multi-Start converge eventualmente para uma solução ótima, mas o tempo esperado para que isso aconteça pode ser muito grande.
·	2.	com 10 cidades quantos vizinhos uma solução tem na vizinhança
·		
	3.	