

1. questão
  - a. questão
  - b. ArrayList: quando o ArrayList chega ao limite do seu tamanho, mas ainda tem há valores que precisam ser inseridos no array, o ArrayList faz uma cópia do ArrayList original com 50% do tamanho do ArrayList original  
LinkedList: quando o LinkedList chega ao limite do seu tamanho, é criada uma Double linked list
  - c. ArrayList: o custo de aumento do tamanho do array quando a capacidade total é preenchida é muito alto,  
LinkedList: o LinkedList possui uma performance maior nos métodos add e remove em comparação ao ArrayList, porém os métodos get e set possuem uma performance pior que a do ArrayList
2. ArrayList 10.000 elementos: 52 Milissegundos  
ArrayList 100.000 elementos: 1461 Milissegundos  
ArrayList 1.000.000 elementos: 158533 Milissegundos  
LinkedList 10.000 elementos: 50 Milissegundos  
LinkedList 100.000 elementos: 883 Milissegundos  
LinkedList 1.000.000 elementos: 140824 Milissegundos
3. Pelo fato de o ArrayList criar uma cópia do dele mesmo quando atinge a sua capacidade máxima tem um grande efeito sobre a performance do programa, na primeira execução tanto o tempo do ArrayList e o tempo do LinkedList tem uma diferença de 2 Milissegundos, porém a medida que o número de elementos inserido no array começa a aumentar a diferença do tempo de cada um começa a ficar maior.
4. Definir um tamanho fixo para o ArrayList, assim não vai ter a necessidade de aumentar o tamanho cada vez que fica com a capacidade máxima