Quinta Lista de Exercícios

1. Implemente um método genérico que apenda todos os elementos de um ArrayList em outro ArrayList. Use um curinga para um dos argumentos de tipo. Forneça duas soluções equivalentes, uma com um ? extends E e uma com ? super E. **(FEITO)**
2. Implemente uma classe Pair<E> que armazena um par de elementos do tipo E. Forneça

Métodos de acesso para obter o primeiro e o segundo elemento. Adicione os métodos max() e min(), para obter o maior e menor dos dois elementos. Para isso, forneça um limitante de tipo apropriado para E. **(FEITO)**

1. Implemente o algoritmo “Sieve of Erathostenes” para determinar todos números primos ≤ n. Para isso, adicione todos os números de 2 a n a um conjunto (Set). Em seguida, encontre repetidamente o menor elemento s do conjunto e remova s2, s · (s + 1), s · (s + 2), e assim por diante. O algoritmo termina quando s2 > n. Implemente usando um HashSet<Integer>. **(FEITO)**
2. Converta todas as strings de um ArrayList<String> para letras maiúsculas. Faça isso com (a) um iterador, (b) um loop sobre os valores de índice e (c) o método replaceAll. **(FEITO)**
3. Escreva um programa que leia todas as palavras de um arquivo txt e imprima com que frequência cada palavra ocorreu. Use um TreeMap<String, Integer>. **(FEITO)**
4. Implemente um método que produza uma tarefa que leia todas as palavras de um arquivo, tentando encontrar uma determinada palavra. Em seguida, para todos os arquivos em um diretório, escalone uma thread para executar a tarefa para cada arquivo. Interrompa todas as outras tarefas quando uma delas tiver sucesso (dica: use invokeAny()). **(FIETO)**
5. Escreva um programa que crie n threads, as quais devem executar tarefas que acumulam valores inteiros em uma variável compartilhada. A threa1 deve acumular 1 na variável compartilhada, a thread2 deve acumular 1+2, a thread3 deve acumular 1+2+3 e a threaN deve acumular 1+2+...+N na variável compartilhada. O acumulador deve ser declarado como public static long acc. Execute o programa várias vezes. Qual o valor do acumulador nas diferentes execuções? Por quê?