EC4070: Data Structures and Algorithms

LAB 05

K.J.M.U.G.S. Eranda Jayasinghe

2021/E/075

SEMESTER 4

EC4070

01.11.2023

Q1.

import java.util.Scanner;

public class CustomHT {

private static final int TABLE\_SIZE = 256;

private CharacterFrequency[] table;

public CustomHT() {

table = new CharacterFrequency[TABLE\_SIZE];

}

public void put(char key) {

int index = key;

if (table[index] == null) {

table[index] = new CharacterFrequency(key);

} else {

table[index].frequency++;

}

}

public CharacterFrequency get(char key) {

int index = key;

return table[index];

}

public static class CharacterFrequency {

char character;

int frequency;

public CharacterFrequency(char character) {

this.character = character;

this.frequency = 1;

}

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a string: ");

String input = scanner.nextLine();

scanner.close();

CustomHT CustomHT = new CustomHT();

for (char c : input.toCharArray()) {

if (Character.isLetterOrDigit(c) || c == ' ' || c == '-' || c == '\_' || c == '?') {

CustomHT.put(c);

}

}

char maxChar = ' ';

int maxCount = 0;

for (int i = 0; i < CustomHT.TABLE\_SIZE; i++) {

CustomHT.CharacterFrequency cf = CustomHT.table[i];

if (cf != null) {

if (cf.frequency > maxCount || (cf.frequency == maxCount && cf.character < maxChar)) {

maxChar = cf.character;

maxCount = cf.frequency;

}

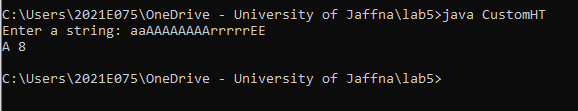
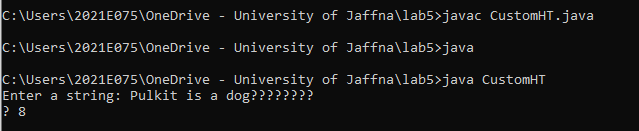
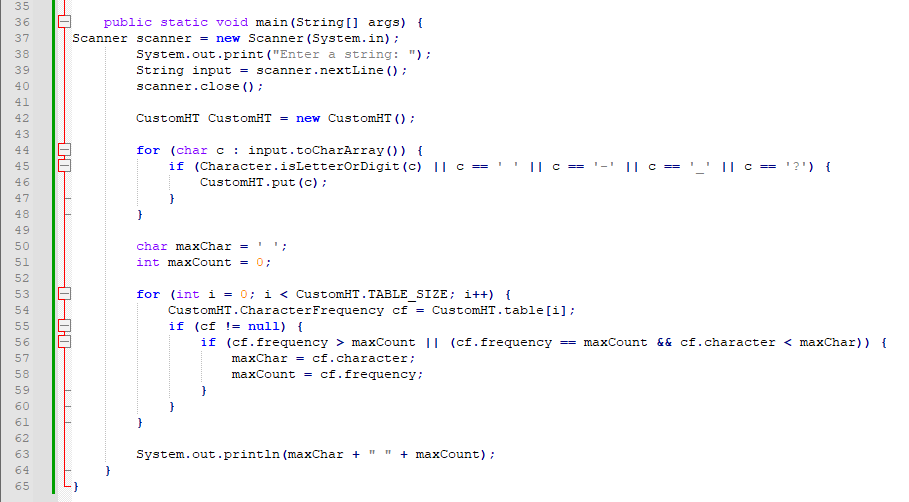
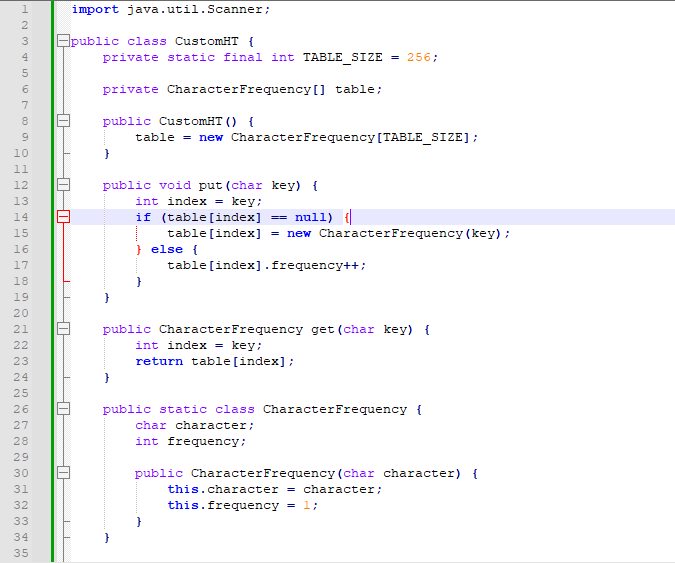
}

}

System.out.println(maxChar + " " + maxCount);

}

}



Q2.

import java.util.\*;

public class FavoriteGame {

private HashMap<String, LinkedList<String>> map;

public FavoriteGame() {

map = new HashMap<String, LinkedList<String>>();

}

public void put(String key, String value) {

if (!map.containsKey(key)) {

map.put(key, new LinkedList<String>());

}

map.get(key).add(value);

}

public int size(String key) {

LinkedList<String> list = map.get(key);

return list != null ? list.size() : 0;

}

int maxElements = 0;

public String getKeyWithMaxElements() {

String keyWithMaxElements = null;

for (Map.Entry<String, LinkedList<String>> entry : map.entrySet()) {

int currentSize = entry.getValue().size();

if (currentSize > maxElements) {

maxElements = currentSize;

keyWithMaxElements = entry.getKey();

}

}

return keyWithMaxElements;

}

public static void main(String[] args) {

FavoriteGame FavoriteGame = new FavoriteGame();

Scanner sc=new Scanner(System.in);

int n;

do{

System.out.println("enter the number");

n=sc.nextInt();

}while(n>=10000 && n<=1);

String name;

String game;

int l;

int i=0;

while(i<n){

do{

System.out.println("enter the name and game");

name=sc.next();

game=sc.next();

l=name.length()+game.length();

}while(l>12);

FavoriteGame.put(game,name);

i++;

}

System.out.println("football "+FavoriteGame.size("football"));

System.out.print(FavoriteGame.getKeyWithMaxElements()+" "+String.valueOf(FavoriteGame.maxElements));

}

}

