class Main{

public static void main(String []args){

    System.out.println("new exo");

    String m1="staple";

    String m2="petals";

    char [] cm1=m1.toCharArray();

    char [] cm2=m2.toCharArray();

     Map<Character,Integer> map=new HashMap<>();

     for(char c: cm1 ){

        if(map.get(c)==null){

            map.put(c,1);

        }

        else{

         map.put(c,map.get(c)+1);

        }

     }

        Map<Character,Integer> map1=new HashMap<>();

     for(char c: cm2 ){

        if(map1.get(c)==null){

            map1.put(c,1);

        }

        else{

         map1.put(c,map1.get(c)+1);

        }

     }//

List<Character> keys=map.keySet().stream().map(e-> e)

.collect(Collectors.toList());

List<Character> keys1=map1.keySet().stream().map(e-> e)

.collect(Collectors.toList());

List<Integer> vals=map.values().stream().map(e-> e).collect(Collectors.toList());

List<Integer> vals1=map1.values().stream().map(e-> e).collect(Collectors.toList());

boolean bk=keys.equals(keys1);

  System.out.println(" boolean keys : "+bk);

boolean bv=vals.equals(vals1);

 System.out.println(" boolean values: "+bv);

//boolean bv=keys.equals(key1);

     Integer p1=0,p2=0;

   for( Integer k:map.values()){

    p1\*=k;

   }

      for( Integer k:map.values()){

    p2\*=k;

   }

  boolean bmap=p1.equals(p2);

  System.out.println(" bmap "+bmap);

  if(p1==p2){

  }

    Arrays.sort(cm1);   System.out.println("cm1 = "+cm1);

    String m1s="",m2s="";

    for(char c:cm1) {

     //   System.out.println(" "+c);

     m1s+=c;

    }

    Arrays.sort(cm2);  System.out.println("cm2 = "+cm2);

        for(char c:cm2) {

     //   System.out.println(" "+c);

     m2s+=c;

    }

    boolean b=m1s.equals(m2s);

    System.out.println(" b = "+b);

}

// Temperature en -273 et 5 en coding game :

https://godbolt.org/z/dad9aTW93

import java.util.ArrayList;  
import java.util.Comparator;  
import java.util.Random;  
import java.util.Scanner;  
  
public class Main {  
 public static void main(String[] args) {  
 System.*out*.println("Hello world!");  
  
 Scanner in = new Scanner(System.*in*);  
  
 *//String sentence = in.nextLine();* ArrayList<Integer> tab=new ArrayList<>();  
 Random rd=new Random();  
 int minimum=-273;  
 int maximum=5526;  
 int n = maximum -minimum + 1;  
  
  
 for(int i=0;i<5;i++){  
 int tr= minimum + rd.nextInt(maximum - minimum + 1);  
 int t = rd.nextInt() % n;  
 int randomNum = minimum + t;  
 tab.add(tr);  
 }  
 System.*out*.println(tab);  
 tab.sort(Comparator.*naturalOrder*());  
  
 int min=tab.get(0);  
  
 System.*out*.println(tab);  
 System.*out*.println(" min : "+min);  
  
  
 int minabs=Math.*abs*(min);  
 System.*out*.println(" min abs : "+minabs);  
 int minf=0;  
 for( Integer e:tab){  
 if( e<minabs){  
 minabs=e;  
 }  
  
  
 }  
  
 System.*out*.println(" min abs final : "+minabs);  
 int maxi=3;  
 int mini=-2;  
 int nx= rd.nextInt(maxi - mini + 1);  
 int trr= mini + nx;  
 System.*out*.println(" random value is : "+nx);  
 System.*out*.println(" radom maxi mini : "+trr);  
}

**Valeur Temp proche de 0**

class Main{

    public static void main(String [] args){

        System.out.println("Hello world !");

 Scanner in = new Scanner(System.in);

 //String sentence = in.nextLine();

 ArrayList<Integer>tab=new ArrayList<>();

 Random rd=new Random();

int minimum=-273;

int maximum=5526;

int n = maximum -minimum + 1;

 for(int i=0;i<5;i++){

    int tr= minimum + rd.nextInt(maximum + minimum + 1);

    int t = rd.nextInt() % n;

    int randomNum =  minimum + t;

    tab.add(tr);

 }

System.out.println(tab);

tab.sort(Comparator.naturalOrder());

int min=tab.get(0);

System.out.println(tab);

System.out.println(" min  : "+min);

int minabs=Math.abs(min);

System.out.println(" min abs : "+minabs);

int indice1=0;

int indiceMin1=0;

for( Integer e:tab){

    int abse=Math.abs(e);

    if( abse<minabs){

         minabs=abse;

        indiceMin1=indice1;

    }

    indice1++;

    }

  System.out.println(" min abs final : "+tab.get(indiceMin1));

List<Integer> tab1=new ArrayList<>(  Arrays.asList(1, -2, -8, 4, 5));

  System.out.println(" tab1 : "+tab1);

tab1.sort(Comparator.naturalOrder());

  System.out.println(" tab1 sorted  : "+tab1);

int min1=tab1.get(0);

int mintab1=Math.abs(min1);

int indice=0;

int indiceMin=0;

for( Integer e:tab1){

    int abse=Math.abs(e);

    if( abse<mintab1){

         mintab1=abse;

        indiceMin=indice;

    }

    indice++;

    }

  System.out.println(" min abs final : "+tab1.get(indiceMin));

    }

}

**Half String**

**Si le mot est impair alors [n/2] donne l’indice du mot au milieu.**

**Et [n/2-1] et [n/2] donne les deux mots aux milieu.**

class Main{

    public static void main(String [] args){

        System.out.println("Hello world !");

 Scanner in = new Scanner(System.in);

 //String sentence = in.nextLine();

 ArrayList<Integer>tab=new ArrayList<>();

 String s="I hate to love you here"; // "I love you khady"; // "I hate to love you here";

 String [] ssplit=s.split(" ");

for(String w :ssplit){

    System.out.print(w+" ");

}

System.out.println(" ");

int n= ssplit.length;

 String ww="";

if(n%2==1){

     ww=ssplit[n/2];

}

else{

 ww=ssplit[n/2-1]+ ssplit[n/2];

}

 System.out.println(" ww : "+ww);

    }

}

**Les algo en fonction ou méthode :**

static fMiddleWordrs(String sentence) **Half String**

class Main{

    public static void main(String [] args){

        System.out.println("Hello world !");

 Scanner in = new Scanner(System.in);

 ArrayList<Integer>tab=new ArrayList<>();

 String s="I hate to love you here"; // "I love you khady"; // "I hate to love you here";

String mw= fMiddleWordrs(s);

System.out.println("fMiddleWordrs(s) : "+mw );  //tolove

    }

 public static String fMiddleWordrs(String sentence){

    String [] ssplit=sentence.split(" ");

    for(String w :ssplit){

    System.out.print(w+" ");

}

System.out.println(" ");

int n= ssplit.length;

 String ww="";

if(n%2==1){

     ww=ssplit[n/2];

}

else{

 ww=ssplit[n/2-1]+ ssplit[n/2];

}

 System.out.println(" ww : "+ww);

    return ww;

  }

}

Si on avait pas mis static fMiddleWordrs(String sentence) l’appel dans le main n’allait pas marcher.

**Valeur plus proche de 0**

https://godbolt.org/z/GYWoc44Te

public static Integer valTempNearToZero(List<Integer> list){

      Stream<Integer> listStream=list.stream().sorted(Comparator.naturalOrder());

      listStream.forEach(e-> System.out.print(e+" "));

    //System.out.println(" ------ ");

      //Stream<Integer> list\_abs= listStream.map(e-> Math.abs(e));

      //list\_abs.forEach(e-> System.out.print(e+" "));

      List<Integer> list2=list.stream()

      .sorted(Comparator.naturalOrder())

      .map(e-> Math.abs(e)).collect(Collectors.toList());

        int indice=0;

        int indiceMin=0;

        Integer min=list2.get(0);

        for(Integer e:list2){

            if(e< min){

                min=e;

                indiceMin=indice;

            }

            indice++;

        }

         Integer min0=list2.get(indiceMin);

          System.out.println( " ");

      System.out.print( " la valeur la  plus proche 0 est : "+min0 );

        return min0}

**IsPalindrome**

**https://godbolt.org/z/EqnnWKP7j**

**Ici Arrays n’a pas de methode reverse. Allors il faut utiliser l’inverse de parcours de boucle avec indice>=0 et et décrementer indice. .**

public static boolean  isPalindrome(String s){

    //char [] tabCar =s.toCharArray();

      String s1="";

      for(int j=s.length()-1;j>=0;j--){

        s1+=s.charAt(j);

      }

      System.out.println(" S1 =  "+s1);

      boolean b=s.equals(s1);

      if(b){

        System.out.println(s+" est Palindrome ");

      }

      else{

        System.out.println(s+" n'est pas  Palindrome ");

      }

     return b;

    }