# Data Structures and Algorithms Exercise 2: E-commerce Platform Search Function:

**Product.java**

package InventoryManagementSystem; public class Product {

int productId;

String productname;

String category;

Product(int productId,String productname,String category){

this.productId=productId; this.productname=productname; this.category=category;

}

}

# Linearsearch.java

package InventoryManagementSystem; import java.util.\*;

public class Linearsearch {

public long linearsearching(Product[] product,String choose) { System.*out*.println("Linear Search");

long start\_time=System.*nanoTime*(); int flag=0;

for(Product i:product) {

if(i.productname.equalsIgnoreCase(choose)) { System.*out*.println("Product Found"); flag=1;

}

}

if(flag==0) {

System.*out*.println("Product not Found");

}

long end\_time=System.*nanoTime*(); long duration=end\_time-start\_time; return duration;

}

}

# Binarysearch.java

package InventoryManagementSystem; import java.util.\*;

public class Binarysearch {

public long binarysearching(Product[] product,String choose) { System.*out*.println("binary Search");

Arrays.*sort*(product, new Comparator<Product>() { public int compare(Product p1, Product p2) {

return p1.productname.compareToIgnoreCase(p2.productname);

}

})

long start\_time=System.*nanoTime*(); int flag=0;

int left=0;

int right=product.length-1; while(left<=right) {

int mid=(left+right)/2;

String name=product[mid].productname;

int num=name.compareToIgnoreCase(choose);

if(num==0) {

System.*out*.println("Product found"); flag=1;

break;

}

else if(num<0) {

left=mid+1;;

}

else {

}

}

right=mid-1;

if(flag==0) {

System.*out*.println("Product not found");

}

long end\_time=System.*nanoTime*(); long duration=end\_time-start\_time; return duration;

}

}

# Main.java

**package** InventoryManagementSystem;

**import** java.util.\*; public class Main {

public static void main(String[] args) {

Scanner sc=new Scanner(System.*in*); Product[] product= {

new Product(1,"SonyTv","Electronics"), new Product(2,"MiTv","Electronics"), new Product(3,"car","vehicle"),

new Product(4,"bus","vehicle")};

System.*out*.println("Enter the product name to search the product");

String choose=sc.nextLine();

System.*out*.println();

linearsearch l=new linearsearch();

long linear\_time=l.linearsearching(product,choose);

System.*out*.println();

binarysearch b=new binarysearch();

long binary\_time=b.binarysearching(product,choose);

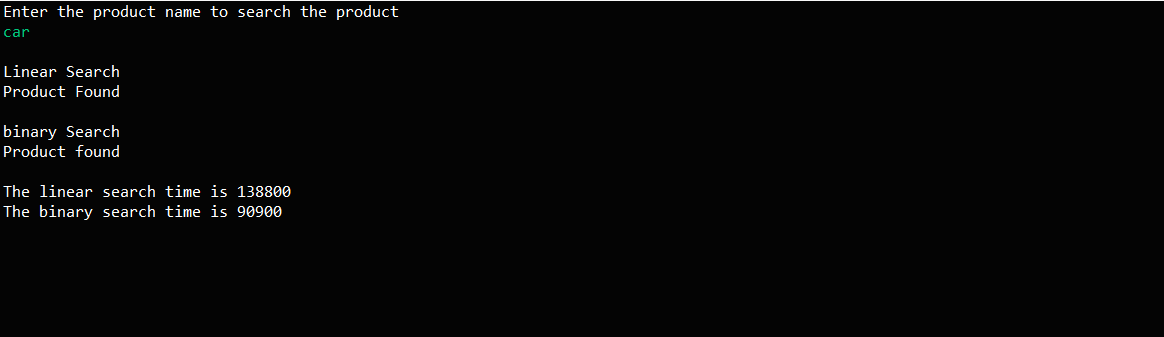
System.*out*.println();

System.*out*.println("The linear search time is "+ linear\_time); System.*out*.println("The binary search time is "+ binary\_time);

}

}

# Output:

****

**Exercise 7: Financial Forecasting:**

**CalculateFutureValue.java**

package FinancialForecasting; import java.util.\*;

public class CalculateFutureValue {

public static double CalculateFV(double pv,double interest,int years) { if(years==0) {

return pv;

}

return *CalculateFV*(pv,interest,years-1)\*(1+interest);

}

public static void main(String[] args) {

Scanner sc=new Scanner(System.*in*);

System.*out*.println("Enter the present value"); double present\_value=sc.nextDouble();

System.*out*.println("Enter the interest"); double interest = sc.nextDouble();

System.*out*.println("Enter the year"); int years=sc.nextInt();

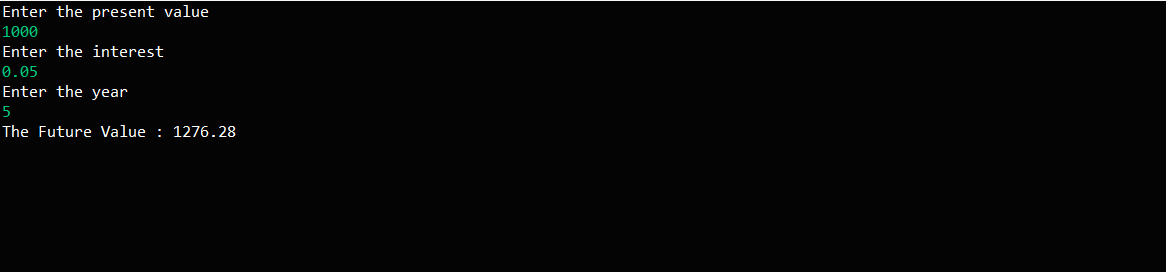
double result=*CalculateFV*(present\_value,interest,years);

System.***out***.printf("The Future Value : %.2f" , result);

}

}

**Output:**

****