

# Lab 03: Basic Object-Oriented Techniques

Đinh Ngọc Khánh Huyền – 20225726

## 1. New written code:

a) Thêm phương thức mới với cùng tên nhưng có tham số kiểu mảng

```
20 public void addDigitalVideoDisc(DigitalVideoDisc[] dvdList)
21 {
22     for (DigitalVideoDisc disc : dvdList) {
23         if (qtyOrdered < MAX_NUMBER_ORDERED) {
24             itemsOrdered[qtyOrdered] = disc;
25             qtyOrdered++;
26             System.out.println("Added: " + disc.getTitle());
27         } else {
28             System.out.println("The cart is almost full. Cannot add: " + disc.getTitle());
29             break;
30         }
31     }
32 }
```

b) Phương thức mới chấp nhận số lượng tham số tùy ý:

```
33 // public void addDigitalVideoDisc(DigitalVideoDisc... dvdList) {
34 //     for (DigitalVideoDisc disc : dvdList) {
35 //         if (qtyOrdered < MAX_NUMBER_ORDERED) {
36 //             itemsOrdered[qtyOrdered] = disc;
37 //             qtyOrdered++;
38 //             System.out.println("Added: " + disc.getTitle());
39 //         } else {
40 //             System.out.println("The cart is full. Cannot add: " + disc.getTitle());
41 //             break;
42 //         }
43 //     }
44 // }
```

c) Phương thức mới với 2 tham số đầu vào:

```
45 public void addDigitalVideoDisc(DigitalVideoDisc dvd1, DigitalVideoDisc dvd2) {
46     if (qtyOrdered < MAX_NUMBER_ORDERED)
47     {
48         qtyOrdered++;
49         itemsOrdered[qtyOrdered-1] = dvd1;
50     }
51 }
52 else
53 {
54     System.out.println("The cart is almost full");
55 }
56 if (qtyOrdered < MAX_NUMBER_ORDERED)
57 {
58     qtyOrdered++;
59     itemsOrdered[qtyOrdered-1] = dvd2;
60 }
61 }
62 else
63 {
64     System.out.println("The cart is almost full");
65 }
66 }
```

d) TestPassingParameter.java

```

1 package AimsProject_class;
2
3 public class TestPassingParameter {
4
5     public static void main(String[] args) {
6         DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
7         DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderella");
8
9         swap(jungleDVD, cinderellaDVD);
10        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
11        System.out.println("cinderella dvd title: " + cinderellaDVD.getTitle());
12
13        changeTitle(jungleDVD, cinderellaDVD.getTitle());
14        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
15    }
16
17    public static void swap(Object o1, Object o2) {
18        Object tmpObject = o1;
19        o1 = o2;
20        o2 = tmpObject;
21    }
22
23    public static void changeTitle(DigitalVideoDisc dvd, String title) {
24        String oldTitle = dvd.getTitle();
25        dvd.setTitle(title);
26        dvd = new DigitalVideoDisc(oldTitle);
27    }
28 }

```

e) Theo dõi số lượng DVD đã tạo

```

1 package AimsProject_class;
2
3 public class DigitalVideoDisc {
4     private static int nbDigitalVideoDiscs = 0;
5     private int id;
6     // Constructors
7     public DigitalVideoDisc(String title, String category, String director, int length, float cost) {
8         super();
9         this.title = title;
10        this.category = category;
11        this.director = director;
12        this.length = length;
13        this.cost = cost;
14
15        nbDigitalVideoDiscs++;
16        this.id = nbDigitalVideoDiscs;
17    }
18
19    public int getId() {
20        return id;
21    }
22 }

```

f) Hàm in ra danh sách đã order trong giỏ hàng, thông tin của từng sản phẩm, tổng giá trị sản phẩm

- toString()

```

65 public String toString() {
66     return "DVD - " + title + " - " + category + " - " + director + " - " + length + ": " + cost + "$";
67 }
68
69

```

- printCart()

```

96 public void printCart() {
97     System.out.println("*****CART*****");
98     System.out.println("Ordered Items:");
99     float totalCost = 0;
100     if (itemsOrdered == null || qtyOrdered <= 0) {
101         System.out.println("Your cart is empty!");
102     } else {
103         for (int i = 0; i < qtyOrdered; i++) {
104             DigitalVideoDisc dvd = itemsOrdered[i];
105             if (dvd != null) {
106                 System.out.println((i + 1) + ". " + dvd.toString());
107                 totalCost += dvd.getCost();
108             }
109         }
110         System.out.printf("Total cost: %.2f $\n", totalCost);
111
112         System.out.println("*****");
113     }
114 }

```

## - Chạy printCart()

The screenshot shows an IDE with a Java file named `AimsProject_class`. The code defines a `main` method that creates a `Cart` object, adds three `DigitalVideoDisc` objects, and calls `printCart()`. The IDE's `Outline` pane on the right shows the `main` method. The `Console` pane at the bottom displays the output of the program.

```

1 package AimsProject_class;
2
3 public class Aims {
4     public static void main(String[] args) {
5
6         Cart anOrder = new Cart();
7
8         DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f);
9         anOrder.addDigitalVideoDisc(dvd1);
10        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star wars", "Science Fiction", "Geogre Lucas", 87, 24.95f);
11        anOrder.addDigitalVideoDisc(dvd2);
12        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladin", "Animation", 18.99f);
13        anOrder.addDigitalVideoDisc(dvd1, dvd2, dvd3);
14
15
16        System.out.println("Total cost is: " + anOrder.totalCost());
17        // anOrder.removeDigitalVideoDisc(dvd1); //Remove successfully dvd1
18        // System.out.println("Total cost is: " + anOrder.totalCost());
19
20        // anOrder.removeDigitalVideoDisc(dvd4); //Not found DVD
21        anOrder.printCart();
22    }
23 }

```

Console Output:

```

Ordered Items:
.. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
!. DVD - Star wars - Science Fiction - Geogre Lucas - 87: 24.95 $
!. DVD - Aladin - Animation - null - 0: 18.99 $
total cost: 63.89 $
*****

```

g) Tìm kiếm DVD theo ID và đưa ra tên các DVD đã tìm được

```

115 // Tìm DVD theo ID
116 public void findbyID(int id) {
117     if (itemsOrdered == null || itemsOrdered.length == 0) {
118         System.out.println("The cart is empty!");
119         return;
120     }
121
122     boolean result = false;
123     for (DigitalVideoDisc dvd : itemsOrdered) {
124         if (dvd != null && dvd.getId() == id) {
125             result = true;
126             System.out.println("Found DVD: " + dvd.toString());
127             break;
128         }
129     }
130
131     if (!result) {
132         System.out.println("Don't found DVD with Id "+id );
133     }
134 }

```

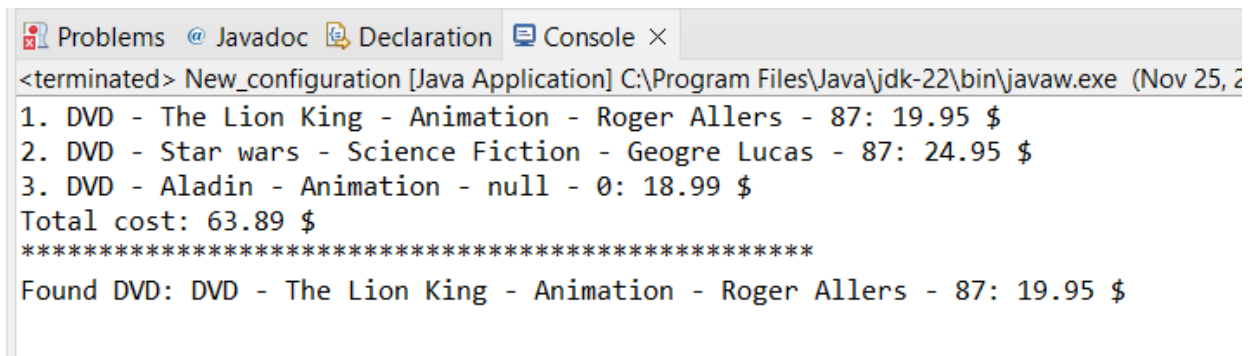
Result:

- ID = 1

```

21 anOrder.printCart();
22 anOrder.findbyID(1);
23

```



```

Problems @ Javadoc Declaration Console ×
<terminated> New_configuration [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (Nov 25, 2
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
2. DVD - Star wars - Science Fiction - Geogre Lucas - 87: 24.95 $
3. DVD - Aladin - Animation - null - 0: 18.99 $
Total cost: 63.89 $
*****
Found DVD: DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $

```

h) Tìm kiếm DVD theo title và đưa ra tên các DVD đã tìm được

- Hàm isMatch(String title)

```

70 public boolean isMatch(String title) {
71     // Kiểm tra nếu title đầu vào hoặc this.title là null
72     if (title == null || this.title == null) {
73         return false;
74     }
75
76     // So sánh không phân biệt chữ hoa/chữ thường
77     return this.title.equalsIgnoreCase(title);
78 }
79

```

- Hàm tìm DVD theo title

```

135 // Tìm DVD theo tên
136 public void findbyTitle(String title) {
137     if (itemsOrdered == null || itemsOrdered.length == 0) {
138         System.out.println("The cart is empty!");
139         return;
140     }
141
142     boolean result = false;
143     for (DigitalVideoDisc dvd : itemsOrdered) {
144         if (dvd != null && dvd.isMatch(title)) {
145             result = true;
146             System.out.println("Found DVD: " + dvd.toString());
147             break;
148         }
149     }
150
151     if (!result) {
152         System.out.println("Don't found DVD with title "+title );
153     }
154
155 }

```

- Result:

Title = Star wars

```

21         anOrder.printCart();
22         anOrder.findbyTitle("Star wars");
23     }

```

```

<terminated> New_configuration [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (Nov 25, 2024, 9:23:21 PM)
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
2. DVD - Star wars - Science Fiction - Geogre Lucas - 87: 24.95 $
3. DVD - Aladin - Animation - null - 0: 18.99 $
Total cost: 63.89 $
*****
Found DVD: DVD - Star wars - Science Fiction - Geogre Lucas - 87: 24.95 $

```

### i) CartTest.java

```

Aims.java  DigitalVideoDisc.java  Cart.java  TestPassingParameter.java  CartTest.java x
1 package AimsProject_class;
2
3 public class CartTest {
4     public static void main(String[] args) {
5         //Create a new cart
6         Cart cart = new Cart();
7
8         // Create new dvd objects and add them to the card
9         DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King","Animation","Roger Allers",87,19.95f);
10        cart.addDigitalVideoDisc(dvd1);
11
12        DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars","Science Fiction","George Lucas",87,24.95f);
13        cart.addDigitalVideoDisc(dvd2);
14
15        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladin","Animation",18.99f);
16        cart.addDigitalVideoDisc(dvd3);
17        //Test the print method
18        cart.printCart();
19        //to-do: test the search methods here
20        cart.findbyID(2);
21    }
22 }
23
Problems  Javadoc  Declaration  Console x
<terminated> CartTest [Java Application] C:\Program Files\Java\jdk-22\bin\javaw.exe (Nov 25, 2024, 9:33:24 PM - 9:33:24 PM) [pid: 15712]
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
2. DVD - Star Wars - Science Fiction - George Lucas - 87: 24.95 $
3. DVD - Aladin - Animation - null - 0: 18.99 $
Total cost: 63.89 $
*****
Found DVD: DVD - Star Wars - Science Fiction - George Lucas - 87: 24.95 $

```

### j) Store.java

```

1 package AimsProject_class;
2
3 public class Store {
4     private static final int MAX_ITEMS_IN_STORE = 20;
5     private DigitalVideoDisc[] itemsInStore = new DigitalVideoDisc[MAX_ITEMS_IN_STORE];
6     private int qtyInStore = 0;
7     // Thêm DVD vào store
8     public void addDVD(DigitalVideoDisc dvd) {
9         if(qtyInStore < MAX_ITEMS_IN_STORE) {
10             itemsInStore[qtyInStore] = dvd;
11             qtyInStore++;
12             System.out.println("Added to store: " + dvd.getTitle());
13         } else {
14             System.out.println("Store is full. Cannot add: "+dvd.getTitle());
15         }
16     }
17     //Xóa DVD khỏi store
18     public void removedDVD(DigitalVideoDisc dvd) {
19         if(qtyInStore == 0) {
20             System.out.println("The store is empty.");
21             return;
22         }
23         for(int i=0; i < qtyInStore; i++) {
24             if(itemsInStore[i] != null && itemsInStore[i].getId()==dvd.getId()) {
25                 for(int j=i; j<qtyInStore-1; j++) {
26                     itemsInStore[j]=itemsInStore[j+1];
27                 }
28                 itemsInStore[qtyInStore-1]=null;
29                 qtyInStore--;
30                 System.out.println("Removed from store: " + dvd.getTitle());
31                 return;
32             }
33         }
34         System.out.println("The DVD " + dvd.getTitle() + " was not found in the store.");
35     }
36     // Hiển thị ds DVD trong Store
37     public void displayStore() {
38         System.out.println("*****STORE*****");
39         if (qtyInStore == 0) {
40             System.out.println("The store is empty!");
41         } else {
42             for (int i = 0; i < qtyInStore; i++) {
43                 System.out.println((i + 1) + ". " + itemsInStore[i].toString());
44             }
45         }
46         System.out.println("*****");
47     }
48 }
49

```

k) StoreTest.java

```

1 package AimsProject_class;
2
3 public class StoreTest {
4     public static void main(String[] args) {
5         Store store = new Store();
6
7         // Create some DVDs
8         DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King", "Animation", "Roger Allers", 87, 19.95f);
9         DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star Wars", "Science Fiction", "George Lucas", 121, 24.95f);
10        DigitalVideoDisc dvd3 = new DigitalVideoDisc("Aladdin", "Animation", 18.99f);
11
12        // Add DVDs to the store
13        store.addDVD(dvd1);
14        store.addDVD(dvd2);
15        store.addDVD(dvd3);
16
17        // Display store items
18        store.displayStore();
19
20        // Remove a DVD
21        store.removeDVD(dvd2);
22
23        // Display store items after removal
24        store.displayStore();
25
26        // Attempt to remove a DVD that does not exist
27        DigitalVideoDisc dvd4 = new DigitalVideoDisc("Frozen");
28        store.removeDVD(dvd4);
29
30        // Display store items after trying to remove non-existing DVD
31        store.displayStore();
32    }
33}

```

- Result:

```

Added to store: The Lion King
Added to store: Star Wars
Added to store: Aladdin
*****STORE*****
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
2. DVD - Star Wars - Science Fiction - George Lucas - 121: 24.95 $
3. DVD - Aladdin - Animation - null - 0: 18.99 $
*****
Removed from store: Star Wars
*****STORE*****
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
2. DVD - Aladdin - Animation - null - 0: 18.99 $
*****
Removed from store: Frozen
*****STORE*****
1. DVD - The Lion King - Animation - Roger Allers - 87: 19.95 $
*****

```

1) String, StringBuilder and StringBuffer:



## - GarbageCreator.java

```
1 package hust.soict.dsai.garbage;
2
3 import java.io.IOException;
4 import java.nio.file.Files;
5 import java.nio.file.Paths;
6
7 public class GarbageCreator {
8     public static void main(String[] args) {
9         String filename = "D:/20241/ooplab/LAB03/OtherProject/src/hust/soict/dsai/garbage/text.txt";
10        byte[] inputBytes = new byte[0];
11        long startTime, endTime;
12
13        try {
14            inputBytes = Files.readAllBytes(Paths.get(filename));
15            startTime = System.currentTimeMillis();
16
17            String outputString = "";
18            for (byte b : inputBytes) {
19                outputString += (char) b;
20            }
21
22            endTime = System.currentTimeMillis();
23            System.out.println("Processing time with String concatenation (using +): " + (endTime - startTime));
24
25            startTime = System.currentTimeMillis();
26
27            StringBuffer sb = new StringBuffer();
28            for (byte b : inputBytes) {
29                sb.append((char) b);
30            }
31            outputString = sb.toString();
32
33            sb = new StringBuffer();
34            for (byte b : inputBytes) {
35                sb.append((char) b);
36            }
37            outputString = sb.toString();
38
39            endTime = System.currentTimeMillis();
40            System.out.println("Processing time with StringBuffer: " + (endTime - startTime) + " ms");
41
42        } catch (IOException e) {
43            System.out.println("Error reading the file: " + e.getMessage());
44        }
45    }
46}
```

## - NoGarbage.java

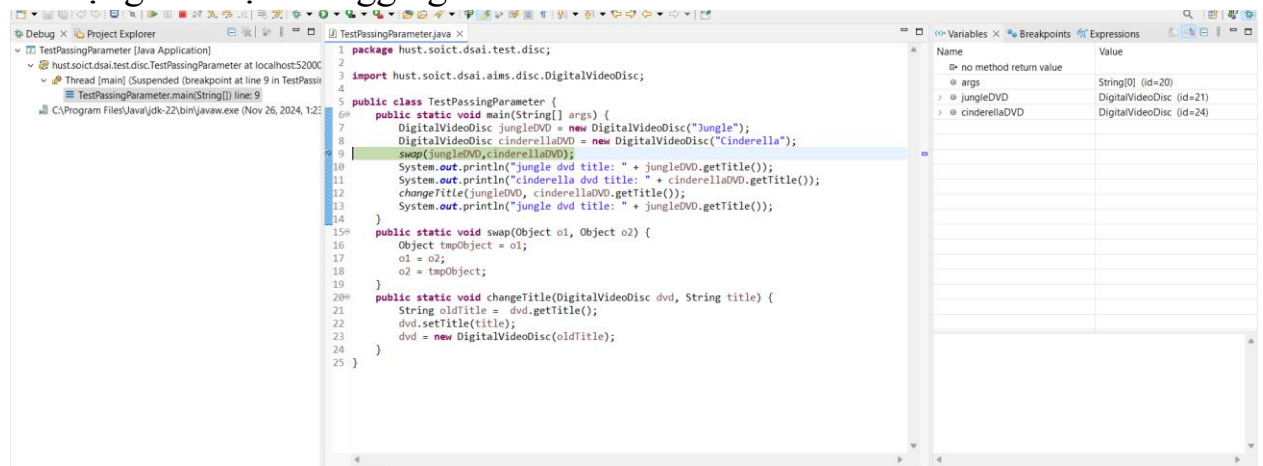
```

1 package hust.soict.dsai.garbage;
2
3 import java.io.*;
4
5 public class NoGarbage {
6     public static void main(String[] args) {
7         StringBuffer sb = new StringBuffer();
8         try {
9             // Read the "largefile.txt" file using StringBuffer to concatenate strings
10            FileInputStream fileInputStream = new FileInputStream("D:/20241/ooplab/LAB03/OtherProject/src/hust/
11            int content;
12            while ((content = fileInputStream.read()) != -1) {
13                sb.append((char) content); // Use StringBuffer to concatenate strings
14            }
15            fileInputStream.close();
16            // Convert the StringBuffer to the final String
17            String result = sb.toString();
18        } catch (IOException e) {
19            System.out.println("Error reading the file: " + e.getMessage());
20        }
21    }
22 }
23

```

## 2. Code debugging and Break-point

Sử dụng chế độ Debugging:



Khảo sát giá trị của biến:

The screenshot shows an IDE with a Java file named `TestPassingParameter.java` and a debug console. The code defines a `DigitalVideoDisc` class and a `TestPassingParameter` class with methods `main`, `swap`, and `changeTitle`. The `main` method creates two `DigitalVideoDisc` objects, `jungleDVD` (title "Jungle", id=25) and `cinderellaDVD` (title "Cinderella", id=33), and calls `swap` and `changeTitle` on them. The debug console shows the state of variables `o1` and `o2` after the `swap` method call. `o1` is a `DigitalVideoDisc` object with title "Cinderella" (id=33) and `o2` is a `DigitalVideoDisc` object with title "Jungle" (id=27). The `changeTitle` method is also shown in the console, with the title of `o2` being changed to "abc" (id=44).

```

1 package hust.soict.dsai.test.disc;
2
3 import hust.soict.dsai.aims.disc.DigitalVideoDisc;
4
5 public class TestPassingParameter {
6     public static void main(String[] args) {
7         DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
8         DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderella");
9         swap(jungleDVD, cinderellaDVD);
10        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
11        System.out.println("cinderella dvd title: " + cinderellaDVD.getTitle());
12        changeTitle(jungleDVD, cinderellaDVD.getTitle());
13        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
14    }
15    public static void swap(Object o1, Object o2) {
16        Object tmpObject = o1;
17        o1 = o2;
18        o2 = tmpObject;
19    }
20    public static void changeTitle(DigitalVideoDisc dvd, String title) {
21        String oldTitle = dvd.getTitle();
22        dvd.setTitle(title);
23        dvd = new DigitalVideoDisc(oldTitle);
24    }
25 }

```

Name	Value
no method return value	DigitalVideoDisc (id=25)
o1	DigitalVideoDisc (id=33)
o2	DigitalVideoDisc (id=22)
tmpObject	DigitalVideoDisc (id=22)

DVD - Cinderella - null - null - 0 min: 0.0 \$

Thay đổi giá trị biến:

The screenshot shows the same IDE with the same Java code, but the `changeTitle` method is now being called with the title "abc" for the `jungleDVD` object. The debug console shows the state of variables `o1` and `o2` after the `changeTitle` method call. `o1` is a `DigitalVideoDisc` object with title "abc" (id=44) and `o2` is a `DigitalVideoDisc` object with title "Cinderella" (id=33). The `changeTitle` method is also shown in the console, with the title of `o2` being changed to "abc" (id=44).

```

1 package hust.soict.dsai.test.disc;
2
3 import hust.soict.dsai.aims.disc.DigitalVideoDisc;
4
5 public class TestPassingParameter {
6     public static void main(String[] args) {
7         DigitalVideoDisc jungleDVD = new DigitalVideoDisc("Jungle");
8         DigitalVideoDisc cinderellaDVD = new DigitalVideoDisc("Cinderella");
9         swap(jungleDVD, cinderellaDVD);
10        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
11        System.out.println("cinderella dvd title: " + cinderellaDVD.getTitle());
12        changeTitle(jungleDVD, cinderellaDVD.getTitle());
13        System.out.println("jungle dvd title: " + jungleDVD.getTitle());
14    }
15    public static void swap(Object o1, Object o2) {
16        Object tmpObject = o1;
17        o1 = o2;
18        o2 = tmpObject;
19    }
20    public static void changeTitle(DigitalVideoDisc dvd, String title) {
21        String oldTitle = dvd.getTitle();
22        dvd.setTitle(title);
23        dvd = new DigitalVideoDisc(oldTitle);
24    }
25 }

```

Name	Value
id	0
length	0
title	"Cinderella" (id=33)
o2	DigitalVideoDisc (id=22)
category	null
cost	0.0
director	null
id	0
length	0
title	"abc" (id=44)
tmpObject	DigitalVideoDisc (id=22)
category	null
cost	0.0
director	null
id	0
length	0
title	"abc" (id=44)

<Choose a previously entered expression>  
abc

Result:

The screenshot shows the IDE's console window with the output of the program. The output is "jungle dvd title: abc".

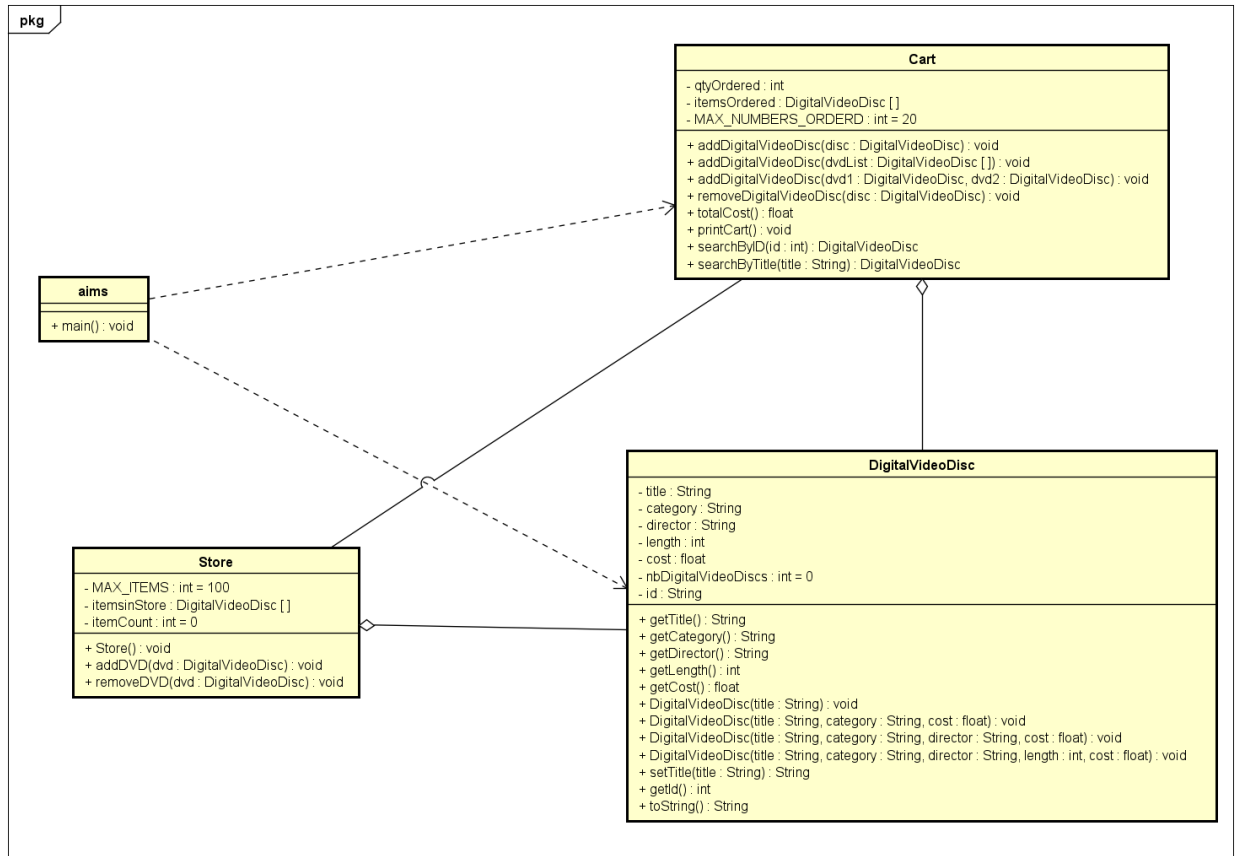
```

TestPassingParameter [Java Application] C:\Program Files\Java
jungle dvd title: abc

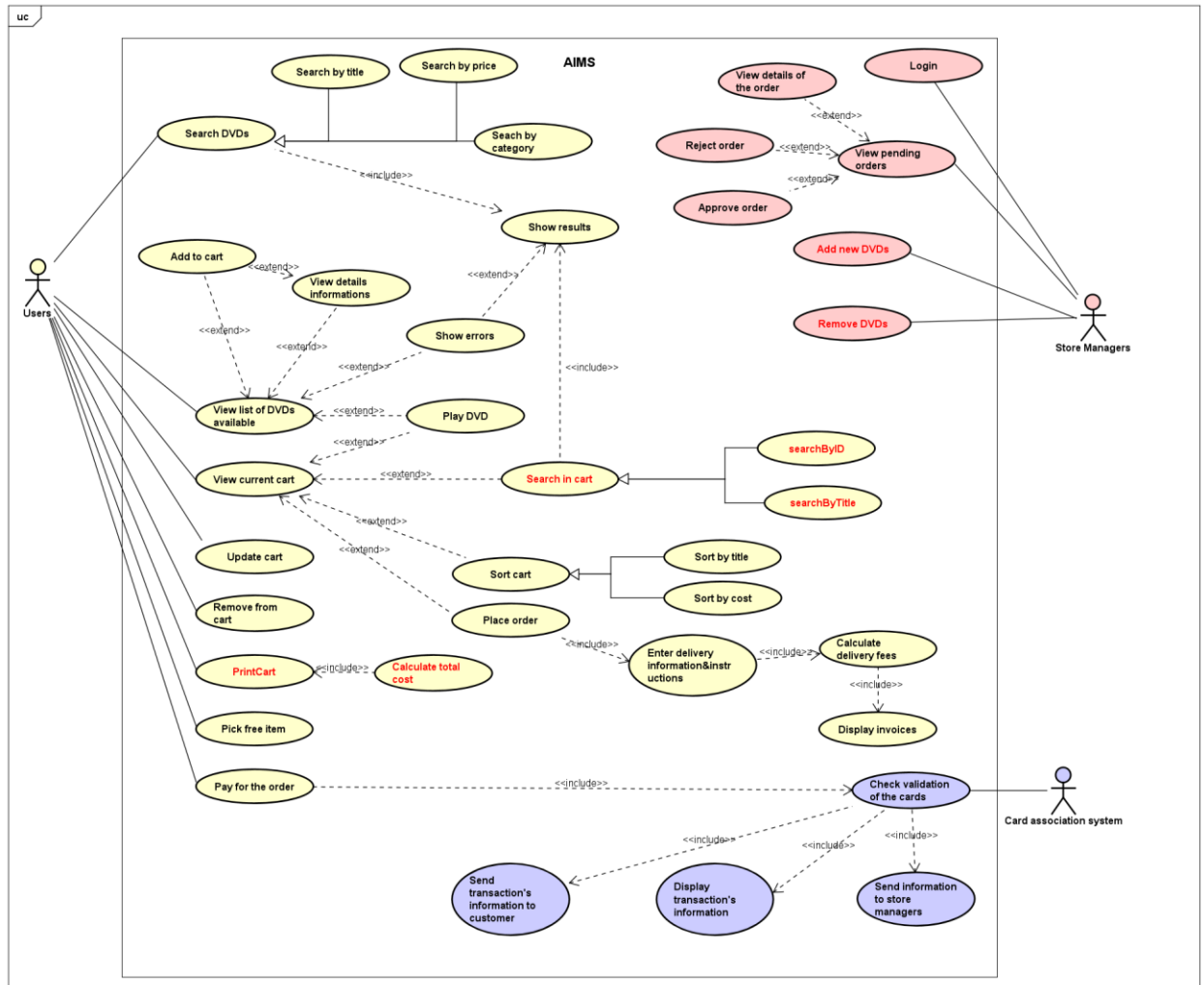
```

### 3. Use-case diagram and Class diagram updates

#### a) Class diagram



#### b) Use-case diagram



**Câu hỏi:**

**Java là ngôn ngữ lập trình truyền tham trị (Pass-by-Value) hay truyền tham chiếu (Pass-by-Reference)?**

**Trả lời:** Java là một ngôn ngữ lập trình truyền tham trị (Pass-by-Value).

**So sánh hai cách sử dụng tham số:**

- **Điểm giống nhau:** Cả hai đều sử dụng vòng lặp bên trong để xử lý các giá trị đầu vào.
- **Điểm khác nhau:**
  - **Tham số mảng:** Yêu cầu phải tạo mảng một cách rõ ràng trước khi truyền vào phương thức, điều này làm giảm tính linh hoạt vì bạn phải khởi tạo mảng trước.

- **Tham số số lượng đối số tùy ý (varargs):** Cho phép truyền trực tiếp nhiều đối số mà không cần tạo mảng, giúp việc gọi phương thức trở nên ngắn gọn và linh hoạt hơn.

**Ý kiến cá nhân:** Trong trường hợp này, em thích sử dụng tham số với số lượng đối số tùy ý vì nó đơn giản và linh hoạt hơn trong quá trình sử dụng.

### 1. Tại sao tiêu đề không thay đổi sau khi gọi swap?

- Trong Java, khi gọi swap(jungleDVD, cinderellaDVD), Java **không truyền tham chiếu của các đối tượng**, mà truyền **bản sao của tham chiếu** (copy of the references).
- Trong phương thức swap, Java chỉ hoán đổi bản sao của các tham chiếu, tức là các bản sao này sẽ trỏ tới các đối tượng trong bộ nhớ. Tuy nhiên, bản thân các tham chiếu gốc (jungleDVD và cinderellaDVD trong main()) không bị ảnh hưởng bởi sự hoán đổi này.
- Vì vậy, sau khi phương thức swap kết thúc, các tham chiếu gốc trong main() vẫn trỏ đến các đối tượng ban đầu. Kết quả là tiêu đề của jungleDVD vẫn là "Jungle" và của cinderellaDVD vẫn là "Cinderella".

### 2. Tại sao tiêu đề của JungleDVD thay đổi sau khi gọi changeTitle?

- Khi gọi changeTitle(jungleDVD, cinderellaDVD.getTitle()), tham chiếu jungleDVD được truyền vào phương thức, và nó cũng là **bản sao của tham chiếu** gốc. Tuy nhiên, điều quan trọng là cả tham chiếu gốc và bản sao đều trỏ đến cùng một đối tượng trong bộ nhớ.
- Khi phương thức changeTitle gọi dvd.setTitle(title), nó thực sự thay đổi **nội dung bên trong đối tượng mà tham chiếu trỏ tới**. Dù dvd chỉ là bản sao của tham chiếu, nhưng vì cả hai tham chiếu đều trỏ đến cùng một đối tượng, thay đổi này được phản ánh trên đối tượng trong bộ nhớ.
- Do đó, tiêu đề của jungleDVD được thay đổi thành giá trị mới, chính là cinderellaDVD.getTitle(), vì chúng ta đã trực tiếp thay đổi dữ liệu trong đối tượng mà jungleDVD trỏ đến.