

Báo cáo thực hành OOP LAB 3

Họ và tên: Phạm Tùng Dương

MSSV: 20225825

Mã lớp: 744520

1. Branch your repository:
2. Working with method overloading:

2.1 Overloading by differing types of parameter:

```
31     public void addDigitalVideoDisc(DigitalVideoDisc[] dvdList) {
32         for (DigitalVideoDisc item : dvdList) {
33             this.addDigitalVideoDisc(item);
34         }
35     }
```

Kết quả :

```
C:\Program Files\Java\jdk-22\bin>java -cp %CLASSPATH% org.javacore.examples.CostCalculator 2008 2009 2010 2011 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022 2023 2024 2025 2026 2027 2028 2029 2030 2031 2032 2033 2034 2035 2036 2037 2038 2039 2040 2041 2042 2043 2044 2045 2046 2047 2048 2049 2050 2051 2052 2053 2054 2055 2056 2057 2058 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 2069 2070 2071 2072 2073 2074 2075 2076 2077 2078 2079 2080 2081 2082 2083 2084 2085 2086 2087 2088 2089 2090 2091 2092 2093 2094 2095 2096 2097 2098 2099 2100 2101 2102 2103 2104 2105 2106 2107 2108 2109 2110 2111 2112 2113 2114 2115 2116 2117 2118 2119 2120 2121 2122 2123 2124 2125 2126 2127 2128 2129 2130 2131 2132 2133 2134 2135 2136 2137 2138 2139 2140 2141 2142 2143 2144 2145 2146 2147 2148 2149 2150 2151 2152 2153 2154 2155 2156 2157 2158 2159 2160 2161 2162 2163 2164 2165 2166 2167 2168 2169 2170 2171 2172 2173 2174 2175 2176 2177 2178 2179 2180 2181 2182 2183 2184 2185 2186 2187 2188 2189 2190 2191 2192 2193 2194 2195 2196 2197 2198 2199 2200 2201 2202 2203 2204 2205 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 2216 2217 2218 2219 2220 2221 2222 2223 2224 2225 2226 2227 2228 2229 2230 2231 2232 2233 2234 2235 2236 2237 2238 2239 2240 2241 2242 2243 2244 2245 2246 2247 2248 2249 2250 2251 2252 2253 2254 2255 2256 2257 2258 2259 2260 2261 2262 2263 2264 2265 2266 2267 2268 2269 2270 2271 2272 2273 2274 2275 2276 2277 2278 2279 2280 2281 2282 2283 2284 2285 2286 2287 2288 2289 2290 2291 2292 2293 2294 2295 2296 2297 2298 2299 2300 2301 2302 2303 2304 2305 2306 2307 2308 2309 2310 2311 2312 2313 2314 2315 2316 2317 2318 2319 2320 2321 2322 2323 2324 2325 2326 2327 2328 2329 2330 2331 2332 2333 2334 2335 2336 2337 2338 2339 2340 2341 2342 2343 2344 2345 2346 2347 2348 2349 2350 2351 2352 2353 2354 2355 2356 2357 2358 2359 2360 2361 2362 2363 2364 2365 2366 2367 2368 2369 2370 2371 2372 2373 2374 2375 2376 2377 2378 2379 2380 2381 2382 2383 2384 2385 2386 2387 2388 2389 2390 2391 2392 2393 2394 2395 2396 2397 2398 2399 2400 2401 2402 2403 2404 2405 2406 2407 2408 2409 2410 2411 2412 2413 2414 2415 2416 2417 2418 2419 2420 2421 2422 2423 2424 2425 2426 2427 2428 2429 2430 2431 2432 2433 2434 2435 2436 2437 2438 2439 2440 2441 2442 2443 2444 2445 2446 2447 2448 2449 2450 2451 2452 2453 2454 2455 2456 2457 2458 2459 2460 2461 2462 2463 2464 2465 2466 2467 2468 2469 2470 2471 2472 2473 2474 2475 2476 2477 2478 2479 2480 2481 2482 2483 2484 2485 2486 2487 2488 2489 2490 2491 2492 2493 2494 2495 2496 2497 2498 2499 2500 2501 2502 2503 2504 2505 2506 2507 2508 2509 2510 2511 2512 2513 2514 2515 2516 2517 2518 2519 2520 2521 2522 2523 2524 2525 2526 2527 2528 2529 2530 2531 2532 2533 2534 2535 2536 2537 2538 2539 2540 2541 2542 2543 2544 2545 2546 2547 2548 2549 2550 2551 2552 2553 2554 2555 2556 2557 2558 2559 2560 2561 2562 2563 2564 2565 2566 2567 2568 2569 2570 2571 2572 2573 2574 2575 2576 2577 2578 2579 2580 2581 2582 2583 2584 2585 2586 2587 2588 2589 2590 2591 2592 2593 2594 2595 2596 2597 2598 2599 2600 2601 2602 2603 2604 2605 2606 2607 2608 2609 2610 2611 2612 2613 2614 2615 2616 2617 2618 2619 2620 2621 2622 2623 2624 2625 2626 2627 2628 2629 2630 2631 2632 2633 2634 2635 2636 2637 2638 2639 2640 2641 2642 2643 2644 2645 2646 2647 2648 2649 2650 2651 2652 2653 2654 2655 2656 2657 2658 2659 2660 2661 2662 2663 2664 2665 2666 2667 2668 2669 2670 2671 2672 2673 2674 2675 2676 2677 2678 2679 2680 2681 2682 2683 2684 2685 2686 2687 2688 2689 2690 2691 2692 2693 2694 2695 2696 2697 2698 2699 2700 2701 2702 2703 2704 2705 2706 2707 2708 2709 2710 2711 2712 2713 2714 2715 2716 2717 2718 2719 2720 2721 2722 2723 2724 2725 2726 2727 2728 2729 2730 2731 2732 2733 2734 2735 2736 2737 2738 2739 2740 2741 2742 2743 2744 2745 2746 2747 2748 2749 2750 2751 2752 2753 2754 2755 2756 2757 2758 2759 2760 2761 2762 2763 2764 2765 2766 2767 2768 2769 2770 2771 2772 2773 2774 2775 2776 2777 2778 2779 2780 2781 2782 2783 2784 2785 2786 2787 2788 2789 2790 2791 2792 2793 2794 2795 2796 2797 2798 2799 2800 2801 2802 2803 2804 2805 2806 2807 2808 2809 2810 2811 2812 2813 2814 2815 2816 2817 2818 2819 2820
```

2.2 . Overloading by differing the number of parameters:

```
23  
24     public void addDigitalVideoDisc(DigitalVideoDisc disc1, DigitalVideoDisc disc2) { no usages new *  
25         this.addDigitalVideoDisc(disc1);  
26         this.addDigitalVideoDisc(disc2);  
27     }
```

Kết quả:

```
24 anOrder.addDigitalVideoDisc(dvd1, dvd2);
25 System.out.println("Total Cost is: " + anOrder.totalCost());
26 }
27 }
```

Run Aims x

"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3\lib\idea_rt.jar=51107:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition 2024.3\bin" -Dfile.encoding=UTF-8

The cart is almost empty
The disc has been added
The disc has been added
The disc has been added
Total Cost is: 69.850006

Process finished with exit code 0

3. Passing parameter:

Question: Is JAVA a Pass by Value or a Pass by Reference programming language?

Java sử dụng cơ chế truyền tham trị, không thực sự truyền tham chiếu như trong C hay C++. Tuy nhiên, khi làm việc với đối tượng, giá trị được truyền chính là tham trị của tham chiếu (địa chỉ bộ nhớ). Điều này có nghĩa là khi truyền một đối tượng như *DigitalVideoDisc* hoặc *Cart* vào một phương thức, Java sẽ sao chép giá trị tham chiếu (địa chỉ bộ nhớ) của đối tượng đó và truyền vào. Nhờ vậy, ta có thể thay đổi các thuộc tính của đối tượng thông qua tham chiếu này. Tuy nhiên, bản thân tham chiếu (địa chỉ bộ nhớ) không thể bị thay đổi. Ví dụ, trong phương thức *changeTitle*, ta có thể sử dụng setter để chỉnh sửa giá trị của một thuộc tính trong đối tượng *dvd*, nhưng không thể thay đổi chính tham chiếu của nó.

Question: After the call of `swap(jungleDVD, cinderellaDVD)` why does the title of these two objects still remain?

- Khi gọi phương thức `swap(jungleDVD, cinderellaDVD)`, giá trị của các tham chiếu (*jungleDVD* và *cinderellaDVD*) được sao chép và truyền vào các tham số `o1` và `o2` của phương thức.
- Bất kỳ thay đổi nào thực hiện trên `o1` và `o2` trong phương thức chỉ tác động đến các bản sao của tham chiếu, mà không ảnh hưởng đến các tham chiếu gốc (*jungleDVD* và *cinderellaDVD*).

Question: After the call of `changeTitle(jungleDVD, cinderellaDVD.getTitle())` why is the title of the *JungleDVD* changed?

Khi truyền một biến có kiểu đối tượng (*object*) vào một hàm, thực chất là ta đang truyền giá trị của tham chiếu mà biến đó lưu trữ, chứ không phải trực tiếp truyền đối tượng mà biến tham chiếu đến. Vì vậy, nếu ta gán biến kiểu đối tượng này cho một tham chiếu khác trong hàm, đối tượng ban đầu mà biến trở đến sẽ không bị ảnh hưởng. Tuy nhiên, nếu ta thao tác trên chính đối tượng được tham chiếu thông qua các phương thức (ví dụ: sử dụng Setter của *Disc*), dữ liệu của đối tượng sẽ thay đổi và thay đổi này vẫn tồn tại sau khi thoát khỏi hàm.

Please write a `swap()` method that can correctly swap the two objects:

```

6 @      public static void swap(DigitalVideoDisc o1, DigitalVideoDisc o2){ 1 usage new *
7         String tmp1 = o1.getTitle();
8         String tmp2 = o1.getCategory();
9         String tmp3 = o1.getDirector();
10        String tmp4 = o2.getTitle();
11        String tmp5 = o2.getCategory();
12        String tmp6 = o2.getDirector();
13        o1.setTitle(tmp4);
14        o1.setCategory(tmp5);
15        o1.setDirector(tmp6);
16        o2.setTitle(tmp1);
17        o2.setCategory(tmp2);
18        o2.setDirector(tmp3);
19    }

```

Kết quả :

```

"C:\Program Files\Java\jdk-1.8\bin\java.exe" ...
jungle dvd title: Cinderella
cinderella dvd title: Jungle
jungle DVD titleJungle

Process finished with exit code 0

```

4. Use debug run:

5. Classifier Member and Instance Member:

```

1 package hust.soict.hedspi.disc;
2
3 public class DigitalVideoDisc { 25 usages 1 nubiz24 *
4     private String title; 6 usages
5     private String category; 5 usages
6     private String director; 4 usages
7     private int length; 3 usages
8     private float cost; 5 usages
9     private int id; 4 usages
10
11     private static int nbDigitalVideoDiscs = 0; 8 usages
12

```

```

28 public DigitalVideoDisc(String title) { 3 usages  nubiz24 *
29     this.title = title;
30
31     nbDigitalVideoDiscs++;
32     this.id = nbDigitalVideoDiscs;
33 }
34 public DigitalVideoDisc(String title, String category, float cost) { 1 usage  nubiz24 *
35     this.title = title;
36     this.category = category;
37     this.cost = cost;
38
39     nbDigitalVideoDiscs++;
40     this.id = nbDigitalVideoDiscs;
41 }
42 public DigitalVideoDisc(String title, String category, String director, float cost) { no usages  nubiz24 *
43     this.title = title;
44     this.category = category;
45     this.director = director;
46     this.cost = cost;
47
48     nbDigitalVideoDiscs++;
49     this.id = nbDigitalVideoDiscs;
50 }
51 public DigitalVideoDisc(String title, String category, String director, int length, float cost) { 2 usages
52     this.title = title;
53     this.category = category;
54     this.director = director;
55     this.length = length;
56     this.cost = cost;
57
58     nbDigitalVideoDiscs++;
59     this.id = nbDigitalVideoDiscs;

```

Kết quả :

```

30 //
31 // System.out.println("Total Cost is: " + anOrder.totalCost());
32 System.out.println("The id of " + dvd1.getTitle() + " is " + dvd1.getId());
33 }
34 }

```

```

"C:\Program Files\Java\jdk-1.8\bin\java.exe" ...
The disc has been added
The id of The Lion King is 1

Process finished with exit code 0

```

6. Open the Cart class:

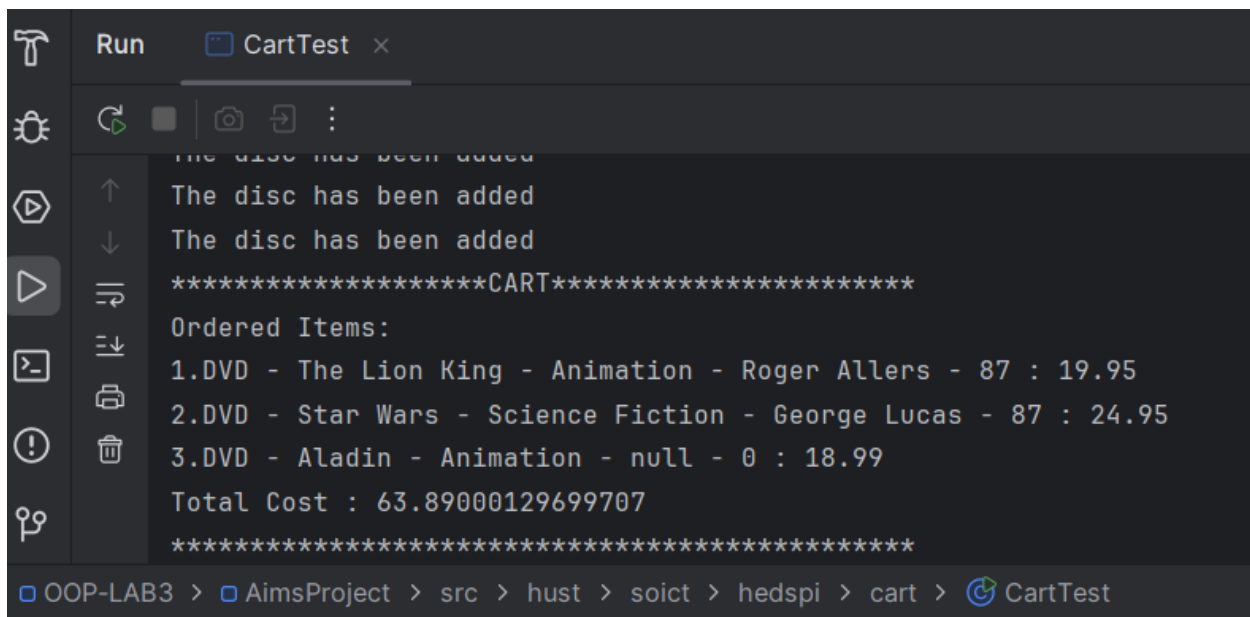
6.1: printCart():

```

public void printCart() { no usages new *
    System.out.println("*****CART*****\nOrdered Items:");
    double total = 0;
    for (int i = 0; i < qtyOrdered; i++) {
        total += this.itemsOrdered[i].getCost();
        System.out.println((i + 1) + ".DVD - " + this.itemsOrdered[i].getTitle() + " - "
            + this.itemsOrdered[i].getCategory() + " - " +
            this.itemsOrdered[i].getDirector() + " - " + this.itemsOrdered[i].getLength() + " : " +
            +this.itemsOrdered[i].getCost());
    }
    System.out.println("Total Cost : " + total);
    System.out.println("*****");
}
}

```

Kết quả trong CartTest:



```

Run  CartTest x
The disc has been added
The disc has been added
*****CART*****
Ordered Items:
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 : 63.89000129699707
*****
OOP-LAB3 > AimsProject > src > hust > soict > hedspi > cart > CartTest

```

6.2: searchCart():

Hàm isMatch():

```

71 public boolean isMatch(int id) { 1 usage new *
72     return this.id == id; // Kiểm tra ID
73 }
74
75 public boolean isMatch(String title) { 1 usage new *
76     return this.title.equalsIgnoreCase(title); // Kiểm tra title (không phân biệt hoa thường)
77 }
78

```

Search by Title:

```

100 // Tìm kiếm theo title
101 public boolean searchByTitle(String title) { no usages new
102     for (int i = 0; i < qtyOrdered; i++) {
103         if (itemsOrdered[i].isMatch(title)) {
104             return true; // Tìm thấy DVD theo title
105         }
106     }
107     return false; // Không tìm thấy DVD theo title
108 }
109 }

```

```

System.out.println("Enter the title of DVD you want to search: ");

```

```

Scanner sc = new Scanner(System.in);
String title = sc.nextLine();

```



```

// Tìm kiếm DVD theo title

```

```

if (cart.searchByTitle(title)) {
    System.out.println("Found DVD with title \"" + title + "\"");
} else {
    System.out.println("Not found DVD with title \"" + title + "\"");
}

```

Kết quả:

```
The disc has been added
The disc has been added
The disc has been added
Enter the title of DVD you want to search:
Aladin
Found DVD with title "Aladin"

Process finished with exit code 0
```

Search by ID:

```
System.out.println("Enter the ID of DVD you want to search: ");

Scanner sc = new Scanner(System.in);
int idForSearch = sc.nextInt();

if (cart.searchById(idForSearch)) {
    System.out.println("Found DVD with ID " + idForSearch);
} else {
    System.out.println("Not found DVD with ID " + idForSearch);
}

sc.close();
```

```
90 // Tìm kiếm theo ID
91 public boolean searchById(int id) { 1usage new*
92     for (int i = 0; i < qtyOrdered; i++) {
93         if (itemsOrdered[i].isMatch(id)) {
94             return true; // Tìm thấy DVD theo ID
95         }
96     }
97     return false; // Không tìm thấy DVD theo ID
98 }
99
```

Kết quả :

```
"C:\Program Files\Java\jdk-1.8\bin\java.exe" ...  
The disc has been added  
The disc has been added  
The disc has been added  
Enter the ID of DVD you want to search:  
5  
Not found DVD with ID 5
```

7. Implement the Store class:

Hàm check trùng lặp:

```
public class Store { no usages  
    private LinkedList<DigitalVideoDisc> itemsInStore = new LinkedList<>(); 5 usages  
  
    private boolean checkDVD(DigitalVideoDisc disc) { 2 usages  
        for (DigitalVideoDisc dvd : itemsInStore) {  
            if (dvd.equals(disc)) {  
                return true;  
            }  
        }  
        return false;  
    }  
}
```

Hàm addDVD():

```
public void addDVD(DigitalVideoDisc disc) { no usages  
    if (!checkDVD(disc)) {  
        itemsInStore.add(disc);  
        System.out.println(disc.getTitle() + " has been added to the store!");  
    } else {  
        System.out.println(disc.getTitle() + " already exists in the store!");  
    }  
}
```

Hàm removeDVD():


```

public void removeDVD(DigitalVideoDisc disc) { no usages
    if (checkDVD(disc)) {
        itemsInStore.remove(disc);
        System.out.println(disc.getTitle() + " has been deleted from the store!");
    } else {
        System.out.println("There is no " + disc.getTitle() + " in the store!");
    }
}
}

```

TEST:

```

import hust.soict.hedspi.disc.DigitalVideoDisc;

public class StoreTest {
    public static void main(String[] args) {
        // Kiểm tra chương trình
        Store store = new Store();

        DigitalVideoDisc dvd1 = new DigitalVideoDisc(title: "Avengers: Endgame");
        DigitalVideoDisc dvd2 = new DigitalVideoDisc(title: "The Matrix");
        DigitalVideoDisc dvd3 = new DigitalVideoDisc(title: "Inception");

        // Thêm DVD
        store.addDVD(dvd1);
        store.addDVD(dvd2);
        store.addDVD(dvd3);

        // Thêm trùng lặp
        store.addDVD(dvd2);

        // Hiển thị cửa hàng
        store.displayStore();

        // Xóa DVD
        store.removeDVD(dvd2);
        store.removeDVD(new DigitalVideoDisc(title: "Avatar"));

        // Hiển thị lại sau khi xóa
        store.displayStore();
    }
}

```

Kết quả :

```
C:\Program Files\Java\jdk-1.8\bin\java.exe ...
```

```
Avengers: Endgame has been added to the store!
```

```
The Matrix has been added to the store!
```

```
Inception has been added to the store!
```

```
The Matrix already exists in the store!
```

```
DVDs in store:
```

```
- Avengers: Endgame
```

```
- The Matrix
```

```
- Inception
```

```
The Matrix has been deleted from the store!
```

```
There is no Avatar in the store!
```

```
DVDs in store:
```

```
- Avengers: Endgame
```

```
- Inception
```

8. Re-organize your projects:

9. String, StringBuilder, and String Buffer:

9.1 Tạo lớp ConcatenationInLoops:

```

1 package hust.soict.dsai.garbage;
2
3 import java.util.Random;
4
5 public class ConcatenationInLoops {
6     public static void main(String[] args) {
7         Random r = new Random( seed: 123);
8         long start = System.currentTimeMillis();
9         String s = "";
10        for (int i = 0; i < 65536; i++) {
11            s += r.nextInt( bound: 2);
12        }
13        System.out.println(System.currentTimeMillis() - start);
14
15        r = new Random( seed: 123);
16        start = System.currentTimeMillis();
17        StringBuilder sb = new StringBuilder();
18        for (int i = 0; i < 65536; i++)
19            sb.append(r.nextInt( bound: 2));
20        s = sb.toString();
21        System.out.println(System.currentTimeMillis() - start);
22    }
23 }
24
25

```

9.2 Tạo lớp GarbageCreator: Chạy thử với file text siêu dài.

163351	nubi
163352	nubi
163353	nubi
163354	nubi
163355	nubi
163356	nubi
163357	nubi
163358	nubi
163359	nubi
163360	nubi
163361	nubi
163362	nubi
163363	nubi
163364	nubi
163365	nubi
163366	nubi
163367	nubi
163368	nubi
163369	nubi
163370	nubi
163371	nubi
163372	nubi
163373	nubi
163374	nubi
163375	nubi
163376	

```

6
7 public class GarbageCreator {
8     public static void main(String[] args) throws IOException {
9         String filename = "C:/Users/Asus/Desktop/OOP-LAB3/OtherProject/src/hust/soict/hedspl/garbage/text.txt";
10        byte[] inputBytes = { 0 };
11        long startTime, endTime;
12
13        inputBytes = Files.readAllBytes(Paths.get(filename));
14        startTime = System.currentTimeMillis();
15        String outputString = "";
16        for (byte b : inputBytes) {
17            outputString += (char) b;
18        }
19        endTime = System.currentTimeMillis();
20        System.out.println(endTime - startTime);
21    }
22 }

```

Kết quả: Chương trình bị treo

9.3 Tạo lớp NoGarbage:

```

public class NoGarbage {
    public static void main(String[] args) throws IOException {
        String filename = "C:/Users/Asus/Desktop/OOP-LAB3/OtherProject/src/hust/soict/hedspl/garbage/text.txt";
        byte[] inputBytes = { 0 };
        long startTime, endTime;

        inputBytes = Files.readAllBytes(Paths.get(filename));
        startTime = System.currentTimeMillis();

        // using StringBuilder
        StringBuilder outputStringBuilder = new StringBuilder();
        for (byte b : inputBytes) {
            outputStringBuilder.append((char) b);
        }
        endTime = System.currentTimeMillis();
        System.out.println(endTime - startTime);
    }
}

```

Kết quả:

```

Run  GarbageCreator  NoGarbage
C:\Program Files\Java\jdk-1.8\bin\java.exe" ...
14
Process finished with exit code 0

```

10. Release flow demonstration:

11. New ClassDiagram:

