

به نام خدا

## تمرین چهارم درس برنامه‌نویسی پیشرفته

نیم‌سال اول ۱۳۹۹-۱۴۰۰

۰. فایل مربوط به توضیحات نحوه ارسال تمرین‌ها را که در مدل قرار دارد، مطالعه کنید.

۱. تمامی فایل‌های کد را به همراه فایل متنی که در قالب pdf است (مورد سوم را بخوانید) به صورت یک فایل آرشیو zip (zip != rar) که به قالب زیر نام‌گذاری شده است، بارگذاری نمایید.

StudentNumber\_FirstName\_LastName.zip

9031066\_Ehsan\_Edalat.zip

۲. در سوال‌هایی که ورودی و خروجی مطلوب آن‌ها مشخص شده است، برنامه‌ی شما به صورت ماشینی تصحیح می‌شود. بنابراین رعایت نحوه ورودی گرفتن و نمایش خروجی اهمیت بسیاری دارد. دقیقاً همان‌طور که از شما خواسته شده است ورودی‌ها را خوانده و خروجی‌ها را تولید کنید.

۳. پاسخ سوالات تشریحی را به صورت تایپ شده و در قالب یک فایل pdf (برای کل تمرین) تحویل دهید.

۴. در صورت مشاهده هرگونه تقلبی، طبق موارد گفته شده در قوانین درس برخورد خواهد شد.

۵. در صورت وجود هرگونه ابهام می‌توانید از طریق ربات تلگرامی [AP\\_Admin\\_bot](#) با تدریس‌یاران در

ارتباط باشید.

مهلت تحویل: تا یکشنبه ۴ آبان ۱۳۹۹ ساعت ۲۳:۵۵ شب

صفحه

## فهرست سوالات

- سوال اول.....۳
- سوال دوم.....۴
- سوال سوم.....۵

## سوال اول

الف) درستی یا نادرستی عبارات زیر را با ذکر دلیل برای عبارات نادرست، تعیین کنید.

۱. در جاوا می‌توانیم آدرس ذخیره شدن یک شی را مشخص کنیم
  ۲. اگر `access modifier` کلاسی `public` نباشد، نمی‌توان از آن کلاس ارث‌بری کرد.
  ۳. در یک کلاس `field`هایی که `protected` تعریف شده‌اند، با استفاده از ارث‌بری، در خارج `package`شان قابل دسترسی‌اند.
  ۴. اگر تابع `override` شده در `subclass`، `public` باشد، می‌توان نتیجه گرفت در `superclass` هم `public` است.
  ۵. متدهای `private` را می‌توانیم `override` کنیم.
- ب) جاهای خالی را با عبارت مناسب پر کنید.
۱. همه‌ی کلاس‌ها در جاوا از کلاس \_\_\_\_\_ ارث‌بری می‌کنند.
  ۲. اگر `field`های یک کلاس را \_\_\_\_\_ تعریف کنیم، `subclass` نمی‌تواند از آن‌ها ارث‌بری کند.
  ۳. وقتی دو تابع در یک کلاس اسم یکسان اما پارامترهای ورودی مختلف داشته باشند، \_\_\_\_\_ رخ داده است.
  ۴. برای `override` کردن یک تابع، تابع نوشته شده در `subclass` باید \_\_\_\_\_ و \_\_\_\_\_ یکسان با آن تابع داشته باشد.

## سوال دوم

به سوالات زیر پاسخ دهید.

۱. فرض کنید کلاس B از کلاس A ارث‌بری می‌کند. توضیح دهید قطعه کد زیر به چه errorی برمی‌خورد؟ به چه دلیل؟ راهی برای رفع این error پیشنهاد دهید.

```
B b1 = new B();  
A a1 = b1;  
b1 = a1;
```

۲. Heap-Stack Diagram را برای قطعه کد زیر وقتی اجرای برنامه به \* می‌رسد، رسم کنید.

```
class A{  
    Integer child = null;  
}  
  
public class Main {  
    public static void main(String[] args) {  
        A ref1 = new A();  
        A ref2 = null;  
        test(ref1, ref2);  
    }  
  
    private static void test(A par1, A par2){  
        par1.child = new Integer("10");  
        par2 = new A();  
        *  
    }  
}
```

## سوال سوم

(پیاده‌سازی)

اصغر به همراه برادرانش، شرکت دیرینه‌ی باربری پدرش را می‌گرداند ولی جدیداً با برادرانش بر سر اداره‌ی شرکت اختلاف نظر پیدا کرده و در تلاش است جای پای خود را محکم‌تر کند. او می‌خواهد برای مشتریانش برنامه‌ای بنویسد تا آنها بتوانند سفارشات خود را ببینند، هزینه‌ی حمل بار را حساب کنند و...، یک برنامه‌ی مدیریت آژانس باربری!

Freight Agency که اسم دیگر این شرکت است، سه نوع خدمات حمل بار را ارائه می‌دهد: زمینی، هوایی و دریایی. در مورد هزینه نهایی ارسال هر بار چندین مورد باید در نظر گرفته شود:








۱. هزینه هر بار، با توجه به کد مبدا و مقصد مشخص می‌شود. به این صورت که اختلاف کد مبدا و مقصد در یک هزینه ثابت (۵۰۰۰ تومان) ضرب می‌شود و هزینه پایه بار مشخص می‌شود.

۲. در مورد وزن بار به ازای هر کیلوگرم بار مبلغ ۲۵۰۰ تومان محاسبه می‌شود (وزن بار عددی اعشاری است که شما باید آن عدد را به سمت بالا گرد کنید)، برای بارهای زیر یک کیلو (صرف نظر از وزن آن) ۲۵۰۰ تومان در نظر گرفته می‌شود. بدیهی است تمام این شروط را باید در متد خود اعمال کنید.

۳. اگر سفارش مشتری‌ای اضافه بار داشته باشد یعنی وزن بار از ۲۰ کیلوگرم بیشتر شود، وزن اضافه در ضریب ۳۰۰۰ تومان ضرب می‌شود. که این مقدار توسط یک تابع جدا (calculateOverload) محاسبه می‌شود و در نهایت با هزینه نهایی جمع می‌شود.

۴. همچنین مشتری می‌تواند بار را بیمه کند، البته هزینه‌ی آن بیشتر می‌شود! در مورد بیمه با توجه به نوع ارسال برای زمینی، هوایی و دریایی به ترتیب مبالغ ۱۰۰۰، ۷۰۰۰ و ۱۲۰۰۰ تومان ضرب در وزن بار می‌شود.

۵. در مورد مبدا و مقصد، تنها در شهرهایی که آژانس شعبه دارد می‌تواند بار را جابه‌جا کند که نام شهرها و کد حرکت از هر شهر در جدول زیر آمده است:

تهران	ساری	بندرعباس	مشهد	کرمان	همدان	قشم	هرمز	شهر
۱۰	۱۱	۱۲	۱۳	۱۶	۱۷	۲۰	۲۱	کد شهر
								امکان ارسال

توجه کنید که مثلاً، کدهای ۲۰ و ۲۱ باربری زمینی ندارند و در صورتی که این کدها به عنوان کدهای مبدا و مقصد برای باربری زمینی داده شود باید پیام مناسب چاپ شود.

۶. در مورد باربری زمینی سه نوع بار وجود دارد: شکستنی، سریع و معمولی. تفاوت این بارها در مقدار هزینه‌ی آنهاست یعنی در مورد بارهای زمینی شکستنی (Ground Breakable) ضریب ۳، زمینی سریع (Ground Fast) ضریب ۲ و زمینی معمولی (Ground Normal) ضریب ۱ در هزینه‌ی آنها اعمال می‌گردد.

۷. می‌توان مشتری‌های جدید اضافه کرد، باید بتوان به حساب هر مشتری سفارشات باربری آنها را اضافه ، سازنده کلاس Client را می‌توانید مطابق نیازتان تغییر دهید(مثلاً برای جست و جو بر اساس نام، می‌توانید فیلد جدید اضافه کنید ولی سعی کنید تا حد ممکن کلاس‌ها را ساده پیاده سازی کنید)

۸. کدی که به شما داده شده است کامل نیست ، شما می‌توانید ورودی‌های تابع‌ها را تغییر دهید و مقدار خروجی تابع را تغییر دهید، اگر تابع دیگری لازم دارید می‌توانید اضافه کنید و محدودیتی در تغییر کد ندارید اما مفاهیم وراثت و چندریختی را از بین نبرید. سطح دسترسی‌ها را نیز باید اصلاح کنید.

راهنمایی : برای محاسبه هزینه نهایی باید در نظر داشته باشید که هزینه ارسال یک محموله در نهایت تابعی است از فاصله، وزن بار، اضافه بار، بیمه بودن یا نبودن (مسئله این است!)، و در ارسال‌های زمینی، نوع ارسال نیز در هزینه نهایی موثر است. متدها و جاوادل‌ها را کامل کنید. متدهای getter و setter اگر لازم دارید اضافه کنید. سوال‌های خود را از طریق بات تلگرام مطرح کنید.

۹. و اما در مورد منو، با اجرا شدن برنامه منو زیر باید به کاربر نشان داده شود:

## 0.Return

- 1.Cargo list
- 2.Client list
- 3.Set a cargo
- 4.Add a client
- 5.Turn off

وارد کردن 0 در این مرحله باید دوباره همین منو را نمایش دهد .

در صورت وارد شدن عدد ۱، باید همه‌ی محموله‌هایی که ثبت شده نشان داده شود، بدیهی است که اگر هیچ باری ثبت نشده باشد باید پیغام مناسب چاپ شود.

## No cargo added yet

بارها باید با شماره و به ترتیب اضافه شدن ، لیست شده باشند. همچنین هزینه کلی، بیمه بودن یا نبودن و کد مبدا و مقصد(به ترتیب) و نوع باربری (اگر زمینی باشد باید نوع آن نیز مشخص شود) برای هر بار باید نمایش داده شود.

1. 34000 insured 12 13 Ground Normal
2. 200000 notInsured 11 13 Sea

...

در صورت وارد شدن عدد ۲، باید لیست تمام مسافران را با شماره نشان دهد:

1. Narges Salehi
2. Sara Salehi

...

پس از آن با وارد کردن **نام هر مسافر** (با وارد کردن عدد صفر باید منو اصلی نمایش داده شود- یک مرحله قبل تر ) باید لیست محموله‌های سفارشی آنها به همراه نام آنها را نمایش دهد(بدیهی است اگر مشتری عضو شده باشد اما سفارشی ثبت نکرده باشد باید پیغام مناسب چاپ شود No cargo set):

## Narges Salehi

1. cargo1

## 2. cargo2

...

در مورد نوع نمایش نام محموله‌ها، برای هر محموله یک فیلد name تعریف کنید و آن‌ها را طوری مدیریت کنید که مطابق نمونه نامگذاری شوند (یعنی باید نام محموله‌ها برای هر مشتری به صورت صعودی و با توجه به اندازه آرایه هر مشتری نامگذاری شود).

در این مرحله کاربر باید بتواند با وارد کردن نام یکی از محموله‌ها، جزییات محموله (مواردی که در لیست یک بود) را مشاهده کند (با زدن شماره صفر باید به لیست محموله‌های آن مسافر بازگردد - یک مرحله قبل تر):

cargo2

34000 insured 12 13 Ground Normal

در صورت وارد شدن عدد ۳ کاربر باید با وارد کردن به ترتیب نام مشتری، کد مبدا، کد مقصد، وزن بار به کیلوگرم، خواستار بیمه بودن با Yes و No، نوع ارسال (اگر زمینی باشد، نوع زمینی بودن ان) یک سفارش را ثبت کند:

Narges Salehi 12 13 30 Yes Ground Fast

Sara Salehi 10 20 58 No Air

Parya Mehrbod 21 20 100 Sea

اگر امکان شیوه ارسال بین کد مبدا و مقصد نباشد باید پیغام مناسب چاپ شود و سفارش ثبت نشود:

Transportation not available

اگر مشتری تا به حال به شرکت سفارشی نداده باشد و در سامانه نباشد، باید آن مشتری را به سامانه اضافه کند، یک پیام چاپ کند و سفارش را ثبت کند:

Client registered

همچنین باید هزینه محموله را به کاربر نشان دهد (چه مشتری عضو بوده باشد و چه نباشد از قبل) نشان دادن هزینه نهایی به منظور ثبت قطعی سفارش است:



Total cost 20000

این محموله باید ثبت شود و در لیست محموله های مشتری و محموله های آژانس قرار بگیرد قرار گیرد.

و با وارد کردن عدد ۴ یک مشتری را تنها با وارد کردن نام به لیست مشتریان اضافه می کنیم (فرض می کنیم هرکسی نام منحصر به فردی دارد) و اگر مشتری از قبل عضو باشد (اسم تکراری) باید پیغام مناسب چاپ شود :

Narges Salehi

Client has already registered

با وارد کردن عدد ۵ هم باید از برنامه خارج شود.

دقت کنید وارد کردن عدد به تنهایی ،گزینه های منو اصلی را اعمال می کند ، در مورد سایر لیست هایی که نمایش داده می شود، مثلا لیست مشتری ها یا لیست سفارشات با وارد کردن نام آنها انتخاب میشود .