

به نام خدا

## تمرین چهارم درس برنامه‌نویسی پیشرفته

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 باشد، بارگذاری نمایید.

StudentNumber\_FirstName\_LastName.zip

9031806\_Mohammad\_Ahmadpanah.zip

۲. در سوال‌هایی که ورودی و خروجی مطلوب آن‌ها مشخص شده است، برنامه‌ی شما به صورت ماشینی تصحیح می‌شود. بنابراین رعایت نحوه ورودی‌گرفتن 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 تایپ‌شده و در قالب یک فایل pdf (برای کل تمرین) تحویل دهید.

مهلت تحویل: تا شنبه ۱۹ اسفند ۱۳۹۶ ساعت ۷:۰۰ صبح

## سوال اول

درستی یا نادرستی هر یک از گزاره‌های زیر را با ذکر دلیل مشخص کنید.

- A) Values of *primitive* types may be stored directly in a collection.
- B) A String object is *immutable*.
- C) A *Set* can contain duplicate values.
- D) A *Map* can contain duplicate keys.

## سوال دوم

مفاهیم زیر را به اختصار توضیح دهید:

Iterator:

Anonymous Object:

Class Variable:

Heap (and its application):

Stack (and its application):

Unit Testing:

## سوال سوم

با استفاده از documentation زبان جاوا، ساختار سلسله مراتب کلاس‌های مشتق شده از Collection را بنویسید. (راهنمایی: پکیج java.util)

## سوال چهارم

طبق برنامه هفتگی مشخص شده، ناهار امروز سلف سرویس دانشگاه شناسل مرغ بود و امروز محمد مدت زمان زیادی را در صف غذاخوری دانشگاه گذراند. در مدت انتظارش، به نوشتن برنامه جاوایی که یک صف را شبیه سازی کند، فکر می کرد تا بتواند به وسیله آن مدت زمان انتظارش را در صف تخمین بزند. حال محمد تصمیم گرفته است از شما برای این کار کمک بخواهد. اگر در مورد صف چیزی نمی دانید یا به غذاخوری دانشگاه مراجعه کنید یا [اینجا](#) را ببینید.

ابتدا شما یک کلاس صف (Queue) پیاده سازی کنید که حداقل متدهای enqueue و dequeue را داشته باشد (راهنمایی: شما می توانید از ArrayList استفاده کنید).

پس از پیاده سازی صف قصد داریم افراد حاضر در آن را شبیه سازی کنیم که بتوانیم زمان انتظار محمد را تخمین بزنیم. به این منظور شما یک کلاس Person بسازید که علاوه بر نام شخص، یک فیلد time

نیز دارد که در constructor مقداره‌ی می‌شود و بیانگر زمان ورود آن شخص به صف (بر حسب دقیقه) است.

حال حداقل ۳۰ نفر را ایجاد کنید. زمان ورود افراد به صف را از صفر شروع کرده و به هر نفر به اندازه‌ی عددی تصادفی (راهنمایی: استفاده از کلاس Random) نسبت به نفر قبل اضافه کنید. در نهایت پس از ساختن صف، تمام آن افراد را از صف خارج کرده و برای هر نفر زمان انتظارش را محاسبه کنید. فرض کنید همه مراحل برداشتن قاشق و سینی، کارت‌زدن و گرفتن غذا برای هر فرد مجموعاً ۲ دقیقه طول می‌کشد و پس از آن، از صف خارج می‌شود. (محاسبه و تخمین زمان پیدا کردن محلی برای نشستن، خود پروژه دیگری را می‌طلبد!)

برای خالی و پر کردن افراد از صف از متدهای enqueue و dequeue که پیشتر طراحی کرده‌اید، استفاده کنید. در طول اجرای برنامه، زمان ورود و خروج هر نفر از صف را چاپ کنید و در نهایت، میانگین زمان انتظار را چاپ کنید.

**توجه!** مستندسازی به کمک JavaDoc، کامنت‌گذاری و رعایت اصول کدنویسی خوانا برای همه کلاس‌های پیاده‌سازی شده الزامی است.

## سوال پنجم

در این سوال، می‌خواهیم اشکال مختلف هندسی را مدل‌سازی کنیم:

کلاس Point بیانگر مدل‌سازی یک نقطه در صفحه مختصات دو بُعدی است. این کلاس را با دو فیلد از نوع مقدار صحیح x و y پیاده‌سازی کنید. در این کلاس متدهایی برای گرفتن و تغییر مقادیر x و y در نظر بگیرید.

کلاس Circle یک دایره را مدل می‌کند که یک نقطه از نوع Point برای مرکز دایره و یک مقدار صحیح به عنوان شعاع دارد.

کلاس Square یک مربع را مدل می‌کند که یک نقطه از نوع Point برای تعیین نقطه شروع (نقطه بالای گوشه سمت چپ یک مربع/مستطیل را نقطه شروع می‌نامیم) و یک مقدار صحیح به عنوان طول ضلع دارد.

کلاس Rectangle یک مستطیل را مدل می‌کند که یک نقطه از نوع Point برای تعیین نقطه شروع و دو مقدار صحیح به عنوان طول و عرض دارد. نقطه شروع این کلاس را static در نظر بگیرید.

کلاس‌های Circle، Square و Rectangle را پیاده‌سازی کنید. همه کلاس‌ها باید دارای همه متدها برای گرفتن و تغییر مقادیر فیلدهای آن باشد. همچنین هر یک از این سه کلاس دو متد برای محاسبه محیط و مساحت آن شکل را دارد. (راهنمایی: استفاده از Math.PI برای عدد پی)

متد main کلاس Run را به این صورت پیاده‌سازی کنید که:

۱- یک نقطه با نام p1 در مختصات (2, 3) قرار داده و دایره‌ای به شعاع 2 حول آن تشکیل دهد. حال با استفاده از متدهایی که برای تغییر x و y نقاط در نظر گرفته بودید، مقدار x و y را برای نقطه‌ی p1 تغییر دهید. آیا مختصات مرکز دایره شما تغییر خواهد کرد؟ **دلیل بیاورید.**

۲- نقطه دیگری به نام p2 در مختصات (0, 0) قرار دهید. حال مربعی به طول ضلع 4 بسازید که نقطه گوشه بالا سمت چپ آن p2 باشد. در صورتی که این نقطه را به p1 تغییر دهیم، مختصات نقطه گوشه بالا سمت چپ مربع تغییر می‌کند؟ آیا Garbage Collector جاوا نقطه p2 را از حافظه پاک می‌کند؟ **چرا؟**

۳- نقطه p3 با مختصات (1, 1) به عنوان نقطه شروع مستطیل‌ها تعیین کنید (این کار را بدون ایجاد نمونه‌ای از کلاس Rectangle انجام دهید). پس از این کار، مستطیل اول را با طول ۵ و عرض ۳، و مستطیل دوم را با طول ۷ و عرض ۶ بسازید. اگر نقطه شروع مستطیل دوم را به p1 تغییر دهیم، چه اتفاقی برای نقطه شروع مستطیل اول خواهد افتاد؟ **چرا؟** آیا پس از این کار، نقطه p2 در حافظه وجود دارد و قابل دسترسی است؟ **دلیل آن چیست؟**

۴- محیط و مساحت همه شکل‌های ساخته‌شده را محاسبه و چاپ کنید.

۵- آیا مربع نوعی مستطیل نیست؟ چگونه می‌توان این مفهوم را در جاوا بیان و مدل کرد؟

**توجه!** مستندسازی به کمک JavaDoc، کامنت‌گذاری و رعایت اصول کدنویسی خوانا برای همه کلاس‌های پیاده‌سازی‌شده الزامی است. همچنین برای این تمرین علاوه بر فایل‌های کد، یک فایل متنی در قالب PDF برای شرح دلایل ارائه کنید.