

به نام خدا

تمرین اول 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 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 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 ۳ اسفند ۱۳۹۸ ساعت ۲۳:۵۵ شب

سوال اول

در مورد تفاوت بین زبان‌های برنامه‌نویسی، کتاب‌خانه‌های زبان و Framework‌ها تحقیق کنید.

سازندگان هر Framework به همراه محصول خود، مستندات یا Documentation‌هایی برای آسان‌تر شدن کار برنامه‌نویسان ارائه می‌دهند. مستندات هر Framework شامل آموزش‌های مقدماتی آن و توضیحاتی دقیق از چگونگی کارکرد قسمت‌های مختلف آن محصول است.

مرجع اصلی زبان جاوا را پیدا کرده و در مورد آن، به صورت مختصر توضیح دهید.

همچنین دو Framework که بر مبنای زبان جاوا پیاده‌سازی شده‌اند و مستندات آن‌ها را پیدا کرده و در مورد هر کدام توضیح مختصری ارائه دهید که مخصوص چه کاری است و روی چه دستگاه‌هایی کار می‌کند.

سوال دوم

در درس مبانی برنامه‌نویسی با نحوه‌ی برنامه‌نویسی ساخت‌یافته (Structured Programming) آشنا شدید. در درس برنامه‌نویسی پیشرفته، با مفاهیم برنامه‌نویسی با دیدگاه شیء‌گرایی (Object-Oriented Programming) آشنا خواهید شد. به انواع مختلف دیدگاه در طراحی و برنامه‌نویسی نرم‌افزارها اصطلاحاً Paradigm (تلفظ کنید : پارادایم) گفته می‌شود. یکی دیگر از پارادایم‌های معروف برنامه‌نویسی، برنامه‌نویسی Functional یا Functional Programming است. با تحقیق به موارد زیر پاسخ دهید:

۱. این ۳ پارادایم را به اختصار شرح دهید.
۲. آیا همین که در یک برنامه از توابع مختلف استفاده کنیم میتوانیم بگوییم که از Functional Programming پیروی کرده ایم؟
۳. به نظر شما تمام برنامه‌هایی که با یک پارادایم (مثلاً OOP) نوشته می‌شوند، می‌توانند در قالب پارادایم‌های دیگر نیز نوشته شوند؟
۴. مزیت برنامه‌نویسی شیء‌گرا نسبت به ۲ پارادایم دیگر چیست؟
۵. معایب برنامه‌نویسی شیء‌گرا نسبت به ۲ پارادایم دیگر چیست؟
۶. آیا پارادایم‌ها در سرعت اجرای برنامه‌ها نقشی دارند؟ اگر بله به چه صورت؟
۷. چند زبان شیء‌گرا را نام ببرید.
۸. آیا در زبان C امکان نوشتن برنامه به صورت شیء‌گرا وجود داشت؟

سوال سوم

برنامه‌ای بنویسید که بتواند تطابق رشته با الگو را بررسی کند. الگو و رشته را می‌توانید در کد به صورت دستی مقدار دهی نمایید.

به برنامه دو عبارت ورودی داده می‌شود که اولی الگو و دومی رشته‌ی مورد سوال است. در صورتی که رشته دوم در قالب الگو می‌گنجد **true** و در غیر این صورت **false** چاپ کنید.

توجه: این برنامه باید به صورت بازگشتی پیاده سازی شود.

توضیح الگوها:

* : به جای این علامت هر زیررشته‌ای می‌تواند قرار گیرد.

! : در صورتی که کاراکتر جایگزین این علامت در رشته، مساوی کاراکتر پیش از ! بود، رشته در الگو می‌گنجد. یا به عبارت دیگر کاراکتر قبل از ! دو بار تکرار شود.

مثال:

Input:

Take*Easy

TakeItEasy

Output:

true

Input:

Take*

TakeItEasy

Output:

true

Input:

*Easy

TakeItEasy

Output:

true

Input:

T*It*y

TakeItEasy

Output:

true

Input:

Take!!tEasy

Take!!tEasy

Output:

true

Input:

Take!!tEasy

TakeItEasy

Output:

false

سوال چهارم

موارد زیر را به اختصار توضیح دهید:

۱. تفاوت میان JDK و JRE
۲. JVM چیست؟ چرا جاوا زبانی است که محدودیت زیادی در مورد دستگاہی که روی آن اجرا می‌شود ندارد؟
۳. اگر سازنده‌ای (constructor) برای یک کلاس تعریف نکنیم، جاوا یک سازنده‌ی پیشفرض برای آن تعریف می‌کند که مقادیر Property های Object ساخته شده را برابر مقادیر پیشفرض هر کدام قرار می‌دهد. مقادیر پیشفرض نوع‌های ابتدایی (Primitive) چیست؟ (۸ مورد)

سوال پنجم

کد زیر را بررسی کرده و طوری تغییر دهید که کامپایل شود:
توجه داشته باشید همه‌ی این کد در یک فایل نوشته شده است.

```
public class SecondClass {  
    private int myInt;  
  
    public void SecondClass (int i){  
        myInt=i;  
    }  
  
    public void getMyInt() { return myInt; }  
}  
public class FirstClass {  
    public static void main(String[]args)  
    {  
        System.out.println("My First class");  
        SecondClass sec = new SecondClass(2);  
    }  
}
```

سوال ششم

جاهای خالی را پر کنید:

۱. کامپایلر جاوا به ازای هر فایل `java`. یک فایل با پسوند ایجاد می کند که قابل تفسیر توسط مفسر جاوا است.
۲. همه کلاس ها در جاوا به طور مستقیم یا غیر مستقیم از کلاس ارث می برند.
۳. تابع `equals(Object o)` در کلاس تعریف شده.
۴. برای مقدار دهی به `property` های یک `object` هنگام ساخته شدن، از استفاده می کنیم.