

به نام خدا

تمرین اول درس برنامه‌نویسی پیشرفته

نیم‌سال دوم ۱۴۰۱-۱۴۰۰

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.

AP-HW1-FirstName_LastName-StudentNumber.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 تایپ‌شده 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 آدرس ایمیل «ap.winter2022@gmail.com» با تدریس‌یاران در ارتباط باشید.

۶. امکان آپلود تا دو روز پس از ددلاین و با ضریب‌های ۷۵ و ۵۰ درصد امکان‌پذیر است.

مهلت تحویل: تا سه‌شنبه ۱۰ اسفند ۱۴۰۰ ساعت ۲۳:۵۹ شب

صفحه

فهرست سوالات

- سوال اول ۳
- سوال دوم ۳
- سوال سوم ۵
- سوال چهارم ۶
- سوال پنجم ۷
- سوال ششم ۸
- سوال هفتم ۹
- سوال هشتم ۱۰
- سوال نهم (امتیازی) ۱۱

سوال اول

به سوالات زیر پاسخ دهید:

- ا. هر یک از اصطلاحات JDK، JRE و JVM را تعریف کرده و تفاوت آن‌ها را بیان کنید.
- ب. تفاوت برنامه‌نویسی ساخت‌یافته^۱ و شیء‌گرا^۲ را توضیح دهید.
- ت. پروسه کامپایل و اجرای یک برنامه جاوا را توضیح دهید. شباهت‌ها و تفاوت‌های آن را با زبان C ذکر کنید.

سوال دوم

* برای جلوگیری از ابهام در صورت سوالات، متن هر سوال به زبان انگلیسی نیز قرار داده شده است.
الف) جاهای خالی را پر کنید.

۱. فایلی که توسط کامپایلر جاوا تولید می‌شود حاوی است که برای اجرای برنامه‌های جاوا تفسیر می‌شود.
۲. حاوی ابزارهای لازم برای نوشتن برنامه‌های جاوا و برای اجرای آن‌هاست.
۳. عملگر کمترین الویت را دارد.
۴. فایل‌های class. حاوی بایت‌کدهای برنامه را می‌گیرد و آن‌ها را به حافظه اصلی منتقل می‌کند.

1. The file produced by the Java compiler contains that are interpreted to execute a Java applet or application.
2. contains tools required to write Java programs, and to execute them.
3. operator has least precedence.

¹ Structural Programming

² Object Oriented

4. The takes the .class files containing the program's bytecodes and transfers them to primary memory

ب) صحیح یا غلط بودن عبارات زیر را مشخص کنید (برای عبارتهای غلط علت نادرستی نیز ذکر شود).

۱. JDK مسئول تبدیل بایت کد به زبان ماشین است.
۲. عملگرها در زبان جاوا از چپ به راست ارزیابی می‌شوند.
۳. زبان برنامه نویسی جاوا به صورت statically-typed است، به این معنی که همه متغیرها قبل از استفاده باید تعریف شوند.
۴. هنگامی که یک برنامه جاوا کامپایل می‌شود، فایل تولید شده توسط کامپایلر با پسوند java ذخیره می‌گردد.
۵. عملگر باقی مانده (%) فقط با عملوندهای صحیح قابل استفاده است.
۶. Null یک کلمه کلیدی (keyword) معتبر در جاوا نیست.

1. JDK is responsible for converting bytecode into machine specific code.
2. Java operators are evaluated from left to right.
3. Java programming language is statically-typed, means all variables should first be declared before they can be used.
4. When a Java program is compiled, the file produced by the compiler ends with the .java file extension.
5. The remainder operator (%) can be used only with integer operands.
6. Null is not a valid keyword in java.

سوال سوم

تابع زیر وظیفه دارد عدد n را به عنوان پارامتر ورودی دریافت کرده و تمامی عوامل اول آن عدد را چاپ کند. این تابع پیاده‌سازی شده است، اما دارای ایراداتی است که عملکرد آن را دچار مشکل و خطا کرده است. ایرادات آن را مشخص و شکل صحیح آن را بنویسید.

```
public class Solution {
    // prints all prime factors of n in order
    public static void printFactors(int n) {
        // find prime numbers from 2 to n
        int primes[] = new int[];
        int primeCount = 0;
        for ( int it = 2; it <= n; it += 1 ) {
            boolean isPrime = true;
            // check if `it` is prime
            for ( int jt = 2; jt * jt <= it; jt += 1 ) {
                if ( it % jt == 0 ) {
                    isPrime = False;
                }
            }
            if ( isPrime == true ) {
                primes[primeCount ++] = it
            }
        }
        // find prime factors of n
        for ( int it = 0; it < primeCount; it += 1 ) {
            if ( n == 1 ) {
                break;
            }
            while n % primes[it] == 0 {
                system.out.print(primes[it] + " ");
                n /= primes[it];
            }
        }
        Systemout.print("\n");
    }
}
```

سوال چهارم

دو مستطیل موازی محورهای مختصات داده شده است. تعیین کنید این دو مستطیل با هم برخورد دارند یا خیر. سپس در صورت برخورد، مساحت مشترک را محاسبه کنید.

ورودی: در دو خط به صورت مجزا مختصات ۴ نقطه هر مستطیل داده شده است که هر دسته با یک اسپیس جدا شده است.

تضمین می شود که مختصات هر نقطه عدد صحیح است.

خروجی: در خط اول در صورت برخورد True چاپ شود و در خط بعدی مساحت مشترک چاپ شود، در غیر این صورت False چاپ شود.

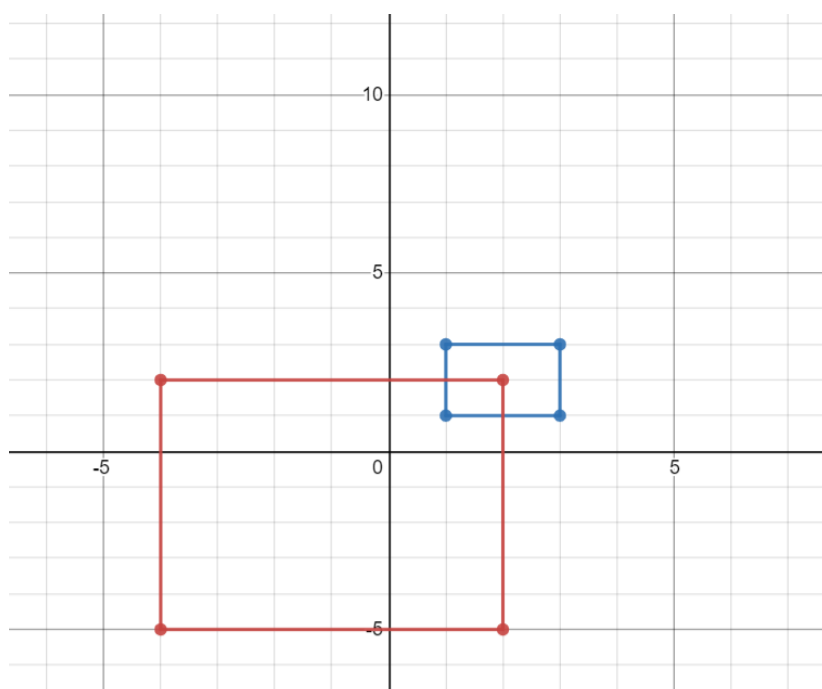
نمونه ورودی:

```
1,1 3,1 1,3 3,3
2,2 2,-5 -4,-5 -4,2
```

خروجی:

```
True
1
```

توضیح: در مختصات داده شده ۴ دسته عدد داده شده که دو به دو راس های مستطیل را تشکیل داده اند که در هر دسته عدد اول مربوط به x و عدد دوم مربوط به y است.



سوال پنجم

برنامه‌ای بنویسید که یک معادله‌ی درجه دوم را در ورودی دریافت کرده و در صورتی که دو ریشه‌ی حقیقی داشت، در خط اول، ریشه‌ی بزرگتر و در خط دوم، ریشه‌ی کوچکتر را نمایش دهد. در صورتی که تنها یک ریشه‌ی حقیقی داشت، در یک خط آن را نمایش دهد و در صورتی که معادله ریشه‌ی حقیقی نداشت، عبارت Roots are not real را چاپ کند. (در صورت اعشاری بودن ریشه‌ها، آن‌ها را تا دو رقم اعشار چاپ نمایید)

توجه داشته باشید در صورتی که مقدار هر یک از ضرایب صفر یا یک باشد، مقدار آن‌ها همچنان نوشته می‌شود و مقدار عبارت سمت راست معادله همواره برابر صفر است.

Input1:

$$(1x^2)+(1x)+(0)=0$$

Output1:

0.0

-1.0

Input2:

$$(1x^2)+(-2x)+(1)=0$$

Output2:

1.0

Input3:

$$(2.2x^2)+(-2x)+(1)=0$$

Output3:

Roots are not real

سوال ششم

برنامه‌ای بنویسید که یک عدد صحیح مثبت n را بگیرد، کوچکترین عدد صحیح بزرگتر از آن را خروجی می‌دهد. خروجی باید دقیقا از همان ارقام تشکیل شده باشد. در صورتی که چنین عددی وجود ندارد، عدد -1 خروجی داده می‌شود. (دقت داشته باشید که استفاده از تابع سورت آماده مجاز نیست!)

Input: 12

Output: 21

Input: 21

Output: -1

Input: 2134

Output: 2143

سوال هفتم

یک pattern و یک string را به عنوان ورودی بگیرید و بررسی کنید که string از الگو تبعیت می کند یا خیر. تبعیت در اینجا به این معناست که هر حرف در الگو دقیقا متناظر دو سویه (bijective) با یک کلمه در string است.

دقت کنید که pattern و string فقط شامل حروف انگلیسی lowercase هستند. در خط اول ورودی یک رشته به عنوان pattern و در خط بعد یک رشته به عنوان string داده می شود، اگر تبعیت میکرد True و در غیر این صورت False چاپ کنید. شکل ورودی و خروجی:

```
Input:
pattern
string
Output:
True/False
```

نمونه های ورودی و خروجی:

```
Input 1:
abca
dog cat fish dog
Output 1:
True
```

```
Input 2:
aaaa
dog cat cat dog
Output 2:
False
```

سوال هشتم

برنامه‌ای بنویسید، که با دریافت عدد صحیح مثبت k ، به صورت بازگشتی، تمام رشته‌های باینری به طول k را که در آن‌ها هیچ دو رقم یک متوالی وجود ندارد، چاپ کند.
دقت داشته باشید که در صورت عدم پیاده‌سازی بازگشتی، نمره‌ای به شما تعلق داده نمی‌شود.

Input: 3

Output: 000, 001, 010, 100, 101

Input: 4

Output: 0000, 0001, 0010, 0100, 0101, 1000, 1001, 1010

سوال نهم (امتیازی)

یک استرینگ باینری متشکل از ۰ و ۱ به شما داده می‌شود. شما می‌توانید آن را به ۳ استرینگ افراز کنید.

(هر سه استرینگ غیر تهی‌اند و $s_1 + s_2 + s_3 = s$ که همان استرینگ اصلی ماست)

شما باید تعداد راه‌هایی که می‌توان s را افراز کرد که تعداد یک‌های s_1 ، s_2 و s_3 برابر باشد را پرینت کنید

● تضمین می‌شود پاسخ نهایی از محدوده int خارج نشود.

Input: 10101

Output: 4

توضیحات:

1 – 010 – 1

10 – 10 – 1

1 – 01 – 01

10 – 1 – 01

Input: 1001

Output: 0

Input: 0000

Output: 3

توضیحات:

0 – 0 – 00

0 – 00 – 0

00 – 0 – 0