

# Ayla Asia Interview Exam A

请回答下列题目。

1. 给定一个正整数 $n$ ，请写一个方法 $f$ 来计算下列表达式并输出结果。注意：请写出最佳性能的解决方案。

$$f(n) = \frac{2}{3 \times 5 \times 7 \times 11 \times 13} + \frac{4}{3 \times 5 \times 7 \times 11 \times 13} + \frac{6}{3 \times 5 \times 7 \times 11 \times 13} + \dots + \frac{2 \times n}{3 \times 5 \times 7 \times 11 \times 13}$$

解答：

```
public static double f(int n) {  
    if(n <=0) return 0;  
    return (f(n-1)+2*n)/(3*5*7*11*13);  
}
```

2. 请写出下列代码的执行结果并解释理由。如果代码无法编译或运行的话，请指出所有错误。

a.

```
public class Test {  
    public class Test2 extends Test {  
        public Test2() {  
            if (n == 0x10)  
                n = 10;  
            else  
                n = 0_10;  
        }  
        public int get() { return n; }  
    }  
  
    int n = 0x10;  
    public Test() { n = 0b10; }  
    public static void main(String[] args) {  
        final Test2 t2 = new Test().new Test2();  
        System.out.println(t2.get());  
    }  
}
```

8

b.

```
public static void main(String[] args) {  
    int x = 5;  
    boolean b1 = true;  
    boolean b2 = false;  
    if ((x == 4) && !b2)  
        System.out.print("1 ");  
        System.out.print("2 ");  
    if ((b2 = true) && b1)  
        System.out.print("3 ");  
}
```

```
}
```

```
2 3
```

```
c.
```

```
public class Test {  
    public static void main(String[] args) {  
        try {  
            get();  
            return;  
        } catch (Exception ex) {  
            System.out.println("Catch");  
        } finally {  
            System.out.println("Finally");  
        }  
        System.out.println("Exit");  
    }  
    public static void get() {  
        System.out.println("Get");  
    }  
}
```

```
Get
```

```
Finally
```

3. 请写一个单例（Singleton）模式（不可用懒汉和饿汉模式，即不可使用Hungry及Lazy模式）。

```
public enum EnumSingleton {  
    INSTANCE;  
    private Object data;  
    public Object getData() {  
        return data;  
    }  
    public void setData(Object data) {  
        this.data = data;  
    }  
    public static EnumSingleton getInstance(){  
        return INSTANCE;  
    }  
}
```

4. 给定一个包含‘a’-‘z’字符的字符串，请写一个方法打印出每个字符，并依照它们的出现次数由少到多来排序。如果有两个字符的出现次数相同的话，请依照字母顺序来排序。例如下列的字符串“eeaddabccccaaeec”，计算过后找出各字母的出现次数为

a: 4次, b: 1次, c: 5次, d: 2次, e: 4次

→ 以出现次数由少到多排序后为

b: 1次, d: 2次, a: 4次, e: 4次, c: 5次

输出结果为“bdaec”。其中b只出现了一次，所以是第一个。a和e都出现了四次，但因为a的字母顺序在e前面，所以先输出a后再输出e。其他例子：

“zzzaccbb” -> “abcz”

“xxxyyz” -> “zyx”

提示：你可以使用Map和Java 8 API来解决。

```
public String strFun(String s) {
    char[] chars = s.toCharArray();
    HashMap<String, Integer> map = new HashMap<>();
    for(int i=0;i<chars.length;i++){
        String str =String.valueOf(chars[i]);
        if(map.get(str) == null ){
            map.put(str, 1);
        }else {
            map.put(str, map.get(str)+1);
        }
    }
    StringBuffer sb = new StringBuffer();
    map.entrySet().stream().sorted(Map.Entry.comparingByValue())
        .forEachOrdered(x -> sb.append(x.getKey()));
    return sb.toString();
}
```

5. 请写一个方法，传入值为一个字符串的List，以及一个字符串，并且回传一个List满足下列要求：

- a. List中的字符串需包含传入的字符串
- b. 每个List中的字符串的内容需要反转
- c. List中的字符串需要去重
- d. List中只需要前三个元素
- e. 请使用**Java Stream** 实现此方法

例如给定一个串列list = [ “not”, “add”, “dog”, “acid”, “add”, “elf”, “gap”, “cat”, “rat” ] 和一个字符串”a”:

规则a → [ “add”, “acid”, “add”, “gap”, “cat”, “rat” ] // 只有包含”a”的元素可留下

规则b → [ “dda”, “dica”, “dda”, “pag”, “tac”, “tar” ] // 反转字符串

规则c → [ “dda”, “dica”, “pag”, “tac”, “tar” ] // 去重

规则d → [ “dda”, “dica”, “pag” ] // 只取前三个元素

→ 输出结果

提示：请使用Java Stream语法，并考虑传入的串列中元素可能为null的情况。

Ans:

```
public List<String> foo(List<String> list, String subString) {  
    return list.parallelStream().filter(Objects::nonNull)  
                .distinct()  
                .filter(str -> str.contains(subString))  
                .limit(3)  
                .map(s -> new StringBuffer(s).reverse().toString())  
                .collect(toList());  
}
```

