Default (1p).

1 (3p). Given the following Java collection:

List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 14,15);

Using Java functional style (Java streams), please write a Java stream program that is doing the following operations in the following order:

   a) Transform each number into a string of characters 'A' whose length is equal to that number
   b) Keep only the string whose length is an odd number

   b) concatenate all the strings

2 (3p). Given the following four classes in Java:

class A implements D {...}   class B extends A implements D {...}   class C extends A implements D {...}

interface D {...}

class Amain{

        ... method1(... list) { return list.get(0);}

       void method2(... list, D el) { list.add(el);}

       void method3(A elem){

            ArrayList<A> listA=new ArrayList<A>(); listA.add(new B());listA.add(new C());

            ArrayList<B> listB = new ArrayList<B>(); listB.add(new B());listB.add(new B());

            ArrayList<C> listC = new ArrayList<C>(); listC.add(new C());listC.add(new C());

            this.method1(listA); this.method1(listB); this.method1(listC);

            this.method2(listA,elem); this.method2(listB,elem); this.method2(listC,elem);

       }

}

Please complete the most specific wildcard types for the class Amain methods (method1 and method2) such that the entire program is correct. **Please justify your solution!** If it is not possible to find a solution please explain the reason. **Without justification you will get 0 points!!!**

3 **(3p).** Is the following Java code correct? Please explain your answer.

```java
class A {

    protected int f1;

    static int s1=0;

    public A(int a)  { this.f1=a*s1;s1=s1+1; }

    static int getS1()  { return s1*getS1(f1); }

    int getS1(int x)  {return (x*s1);}

}
```