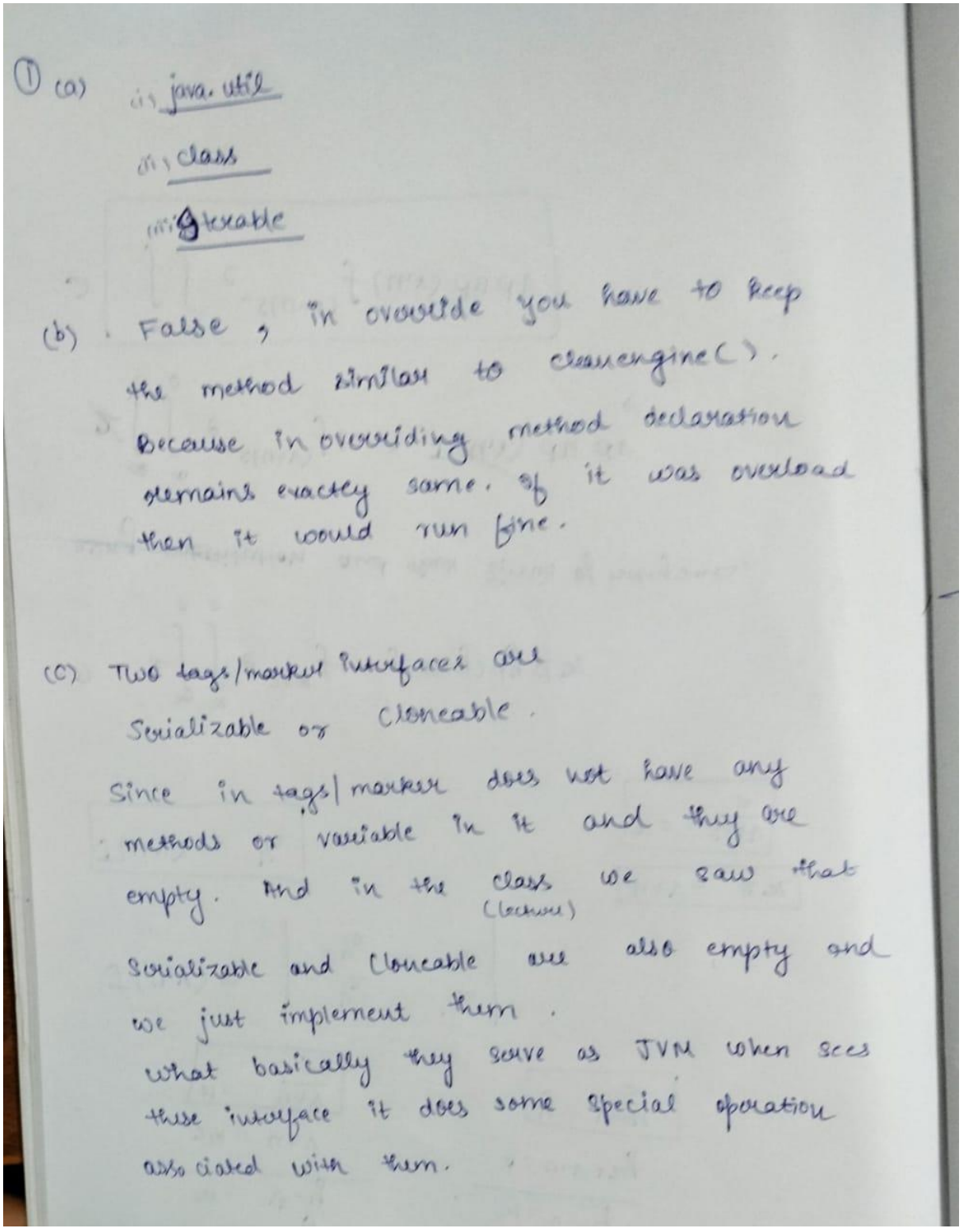
Name – Vibhu Jain

Roll No – 2020151



① (a) (i) java.util

(ii) class

(iii) Iterable

(b) False ; in override you have to keep the method similar to cleanengine(). Because in overriding method declaration remains exactly same. If it was overload then it would run fine.

(c) Two tags/marker interfaces are Serializable or Cloneable.

Since in tags/marker does not have any methods or variable in it and they are empty. And in the class (lecture) we saw that Serializable and Cloneable are also empty and we just implement them.

what basically they serve as JVM when sees these interface it does some special operation associated with them.

(d) By using Polymorphism (compile-time) we can achieve that since the compiler knows about the method we are calling exactly.

Advantages:

Main advantage is it gives user a way to reuse code. We just have to declare a method and can assign the code in other like as in interfaces.

2) (a)

```
final class Point2D <T> {
    private T X-co-ordinte;
    private T Y-co-ordinate;
    public Point2D ( T x, T y )
    {   X-co-ordinate = x;
        Y-co-ordinate = y; }
    public T getX() { return X-odinate;}
    public T getY() { return Y-ordinate; }

public class Main {
    public static void Main (String args[]))
{   Point2D p = new Point2D ( 10, 16);
    p.getX();
    p.getY();
}
```

(2)

(b)

```
public class Fibonacci extends RecursiveTask <Int> {
private static Map <Int, Int> mapping =
                new Hashmap <Int, Int> ();
  int n ;
public Fibonacci (int n) { n = n ; }
public Integer compute () {
   if (n<2) return n ;
if (mapping. containsKey (n))
        return (mapping. getValue (n))
   else
      int result = //-------- //
mapping. setValue (n);
      return result ; }
public static void main (Strings [] args)
{
   // . . . . . //
}
```

Q.3

(a) Abstract class does not have all its method abstract whereas in Interface all the methods are abstract only.

We use abstract keyword for abstract class and interface for interface class.

Interface usually supports multiple inheritance while abstract class does not support it.

(3)

(b)