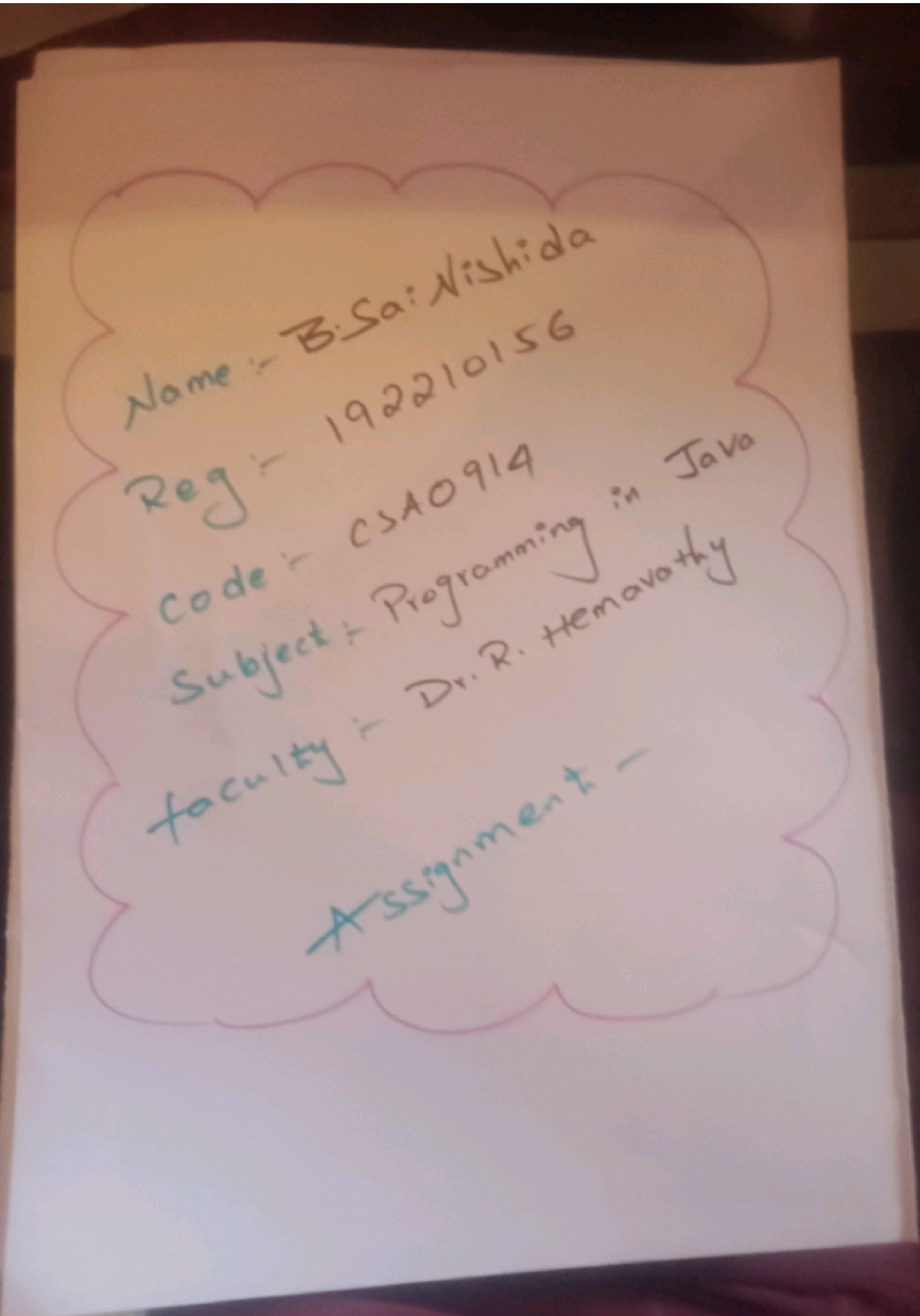

Name :- B. Sai Nishida

Reg :- 192210156

code :- CSA0914

Subject :- Programming in Java

faculty :- Dr. R. Hemavathy

Assignment -

① Array operations

Aim: To write Program on array operations

Pseudo code:

```
Initialize an Empty arraylist (string)
Add elements ("Apple", "Banana", "cherry", "Date")
 to the Arraylist
-> specify an index to remove (index 2)
-> Remove the element at the specified index
   from the Arraylist
-> specify the element to search
-> use indexof method to find the position
-> Print position
-> use a loop to iterate the Arraylist
-> Print element
```

Code:

```
import java.util.Arraylist;
Public class ArraylistOperations {
   Public static Void main(string[] args){
    Arraylist<String> list = new Arraylist<>();
     list.add("Apple");
     list.add ("Banana");
     list.add ("cherry");
     list.add ("Date");
    system.out.println("Initial List: "+list);
    int indexToRemove = 2;
if (indexToRemove >=0 && indexToRemove <list.size()){
  list.remove (indexToRemove);
  System.out.println ("List after removing:");
 for(string element: list){
      Systemout.println(element);
   }
 }
}
```

Output:

```
Initial List: [Apple, Banana, Cherry, Date]
List after removing element at index 2:"
  [Apple, Banana, Date]
The position of "Date" is: 2
All elements in the list:
Apple
Banana
Date
```

② Hash Set Operation

Aim: write Program on HashSet operation

Pseudo code

```
-> Declare new HashSet with names [John, Alice, Bob, Alice]
-> Add names in names
-> create Boolean is_present
-> declare & use function names.contain( "Bob")
```

```
->Print "Present"
->Initialize for each loop with string name : names
  Print "name'
 end loop
->end Program.
```

code:

```
import java.util.Hashset;
Public class Hashoperations {
 Public static void (string [] args){
 Hashset<string> names = new Hashset<>();
    names.add("John");
    names.add(" Alice");
    names.add(" Bob");
    names.add(" Alice");
boolean is present = names.contains("Bob");
 System.out.println("is Bob Present:"+ is
                         Present");
 for(string name ; names)
 {
 System.out.println(name);
 }
 }
}
```

O/P

```
is Bob Present : is present
John
Alice
Bob
Alice
```

(3) Priority Queue

Aim:- To write Program on Priority queue

Pseudo code

```
=> Declare new Queue (string) with name employees
=> Add elements [Alice, John, Eve -]
=> Employee. poll();
=> initialize for each loop:-
        String employee : Employees
        Print ("Employee");
      }
    }
  }
```

Code :-

```
import java.util.priorityQueue;
  Public Class Priority{
   Public Static Queue <string> Employees =
              new Priority Queue ( );
   employeees. add ("Alice");
   employees add ("John");
    Employees. add ("Eve");
  System. out. println ("Removed highest priority:").
    employees. poll());
for(string Employee : Employees){
 System. out. println (employees);
}
```

(4) HashMAP

Aim:- To write HashMap Program in java

Pseudo code:-

```
→ Initializ new Hashmap (string) students
→ insert [ John , Alice , Bob] with inden [101,102,103]
→ Use for each loop
→ Print "ID:"+ entry.getKey()+", Name :"+
          entry.getValue();
```

Code:-

```
import Java.util.HashMap;
Public class HashMapDemo{
 Public static void main (String[] args){
  HashMap<Integer, String> students = new
                  HashMap<>();

  Students.put (101,"John");
   Students.Put (102," Alice");
   Students.Put (103," Bob");
 String name= student.get(102);
 System.out.Println ("Student with ID 102"+
          name);
 students.remove (103);
 for(HashMap.Entry <Integer, string entry>
              Students.entry()){
  System.out.Println(" ID:"+ entry.getKey()+
        ",name:"+ entry.getValue());
      }
    }
  }
```

O/P:

```
Student with ID 102 : Alice
101 : John
 102 : Alice.
```