Question 1: NADRA is a Pakistani agency responsible for maintaining records of Pakistani citizens by entering their information in the database. It then issues a unique ID card to every registered citizen as a proof of successful registration.

There are some mandatory steps for this registration process.

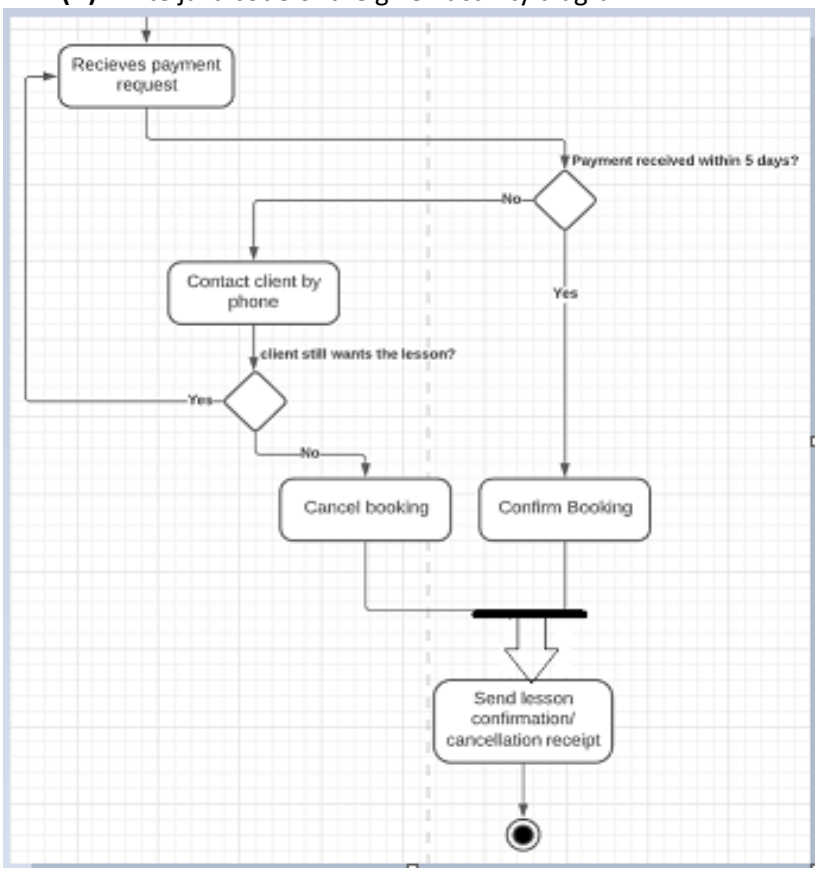
First of all, when a customer enters the NADRA Office, he/she gets his/her token from the kiosk system and then the following main issuance service begins and its steps are documented as follows:

1)Input the B-form details from the applicant. Also verify the B-Form from the system.
2)Take current bio data of the applicant as input.
3)Capture digital fingerprints of the applicant.
4)Capture the digital signature of the applicant.
5)Nadra officers should be able to capture pictures of the applicant and store them in the system.
6)Generate a receipt containing information about the expected issue date of their ID card.
7)Deliver the ID card at the specified location.

You have to design an **Activity diagram** for this process. Keep following things in mind while making the diagram:

- Include expansion regions.
- Mention the appropriate keyword of expansion region along with the reason of your choice.
- Identify and include basic conditions of ID card issuance.
- Identify objects and show them using swim lanes.
- Model exceptions and their types in case of any error or alternate behavior.
- Use interruptible activity regions as well.

Question: 2 (A) Write java code of the given activity diagram.



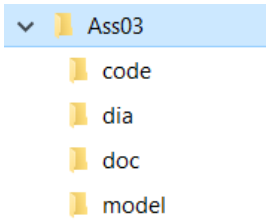
Question: 2 (B) Draw activity diagram from the given code

```
package Loops;

import java.util.Scanner;

public class DoWhileLoop {
        private static Scanner input;

        public static void main(String[] args) {
                int number, count = 0;
                input = new Scanner(System.in);

                System.out.println("\n Please Enter any integer ");
                number = input.nextInt();

                do  {
                        if(number>10){
                                count= count + number+2;
                        }
                                if(number>10 && count<100){
                                count= count +4;
                        }
                        count= count + number;
                        number--;
                } while (number > 0);
                System.out.format(" Sum of the Numbers is: %d ", count);
        }
}
```

Question: 2 (C) Draw activity diagram from the given code

```
1
2  import java.util.Calendar;
3  import java.util.GregorianCalendar;
4
5  class MyFriend {
6
7      protected GregorianCalendar born;
8      protected String name;
9
10     public void Friend(String name, GregorianCalendar born) {
11
12         this.name = name;
13         this.born = born;
14     }
15
16     public void info() {
17
18         System.out.format("%s was born on %s/%s/%s\n",
19                 this.name, this.born.get(Calendar.DATE),
20                 this.born.get(Calendar.MONTH),
21                 this.born.get(Calendar.YEAR));
22     }
23 }
24
25 class MyFriendFather extends MyFriend {
26
27     private GregorianCalendar fborn;
28     private String fname;
29
30     public void FriendFather(String fname, GregorianCalendar fborn) {
31
32         this.fname = fname;
33         this.fborn = fborn;
34     }
35
36     public void info(String name,GregorianCalendar born) {
37         MyFriend f=new MyFriend();
38         f.name=name;
39         f.born=born;
40          System.out.format("%s was born on %s/%s/%s\n",
41                 f.name, f.born.get(Calendar.DATE),
42                 f.born.get(Calendar.MONTH),
43                 f.born.get(Calendar.YEAR));
44         System.out.format("My Friend's Father, Mr %s was born on %s/%s/%s\n",
45                 this.fname, this.fborn.get(Calendar.DATE),
46                 this.fborn.get(Calendar.MONTH),
47                  this.fborn.get(Calendar.YEAR));
48     }
49 }
50 public class MemberInit {
51
52     public static void main(String[] args) {
53
54         String name = "Lenka";
55         GregorianCalendar born = new GregorianCalendar(1990, 3, 5);
56
57         MyFriend fr = new MyFriend();
58         fr.Friend(name,born);
59         fr.info();
60
61           String fname = "Benka";
62         GregorianCalendar fborn = new GregorianCalendar(1980, 3, 5);
63
64         MyFriendFather ffr = new MyFriendFather();
65         ffr.FriendFather(fname,fborn);
66         ffr.info(name,born);
67     }
68 }
```

Important Note:



1. Date assigned 05-11-2019. Last date of submission is **18-11-2021 11 AM sharp.**
2. Assignments will not be accepted after due date.
3. Students are required to submit the assignment individually.
4. Plagiarism, if detected, will result in zero marks.
5. Assignment must be submitted via slate. Only one submission is allowed.
6. Submit the assignment after making a single zip archive of the assignment files.
7. Folder hierarchy: MS Word document report in /doc folder, generated source code in /code folder (if any), all exported PNG diagrams in /dia folder and Papyrus models and project files in /model directory are required. (See the directory structure as shown in figure above). Use only Papyrus for modeling and submit the project files.
8. Archive the assignment and name it “FASTAssign03YourRollNo.zip”
9. Cover Page of Assignment document must contain: Student name, Roll no, Date of submission.
10. Do not submit assignment on 11^{th} hour, always submit assignment 3-4 hours or one day earlier then deadline, if you do it on 11^{th} hour, then due to connectivity issues you may not be able to submit the assignment on time.
11. Deadline will not extend.