Set B Part 1

Design Marks: Total = 7

store_details(n, A, B)

// Array A and B of size *n* initialised into -1.

```
1. read the value of s                //number of students
2. initialise an array C of size m //to track the positions of students in the
   order of reg_no
3. j = 0  //keep track of index in array C
4. for i = 1 to s
   do read  reg_no                    //register number
      read m                          //mark
      compute $p = (m+10) \bmod n$      //find position p
     do
          if A[p] = -1   //position p is vacant
            //assignreg_no, m into position p of array A and B respectively
            then    A[p]   =  reg_no
                    B[p]   =  m
                    C[j++] =  p
          else
                 //find k which is the next vacant position
                 for pos  = 1 to n
                        do
                           compute $k = (p+pos) \bmod n$
                           if A[k] = -1
                               then
                                      A[k]   =  reg_no
                                      B[k]   =  m
                                      C[j++] =  k
                                      break   // break out of  for

5. for i = 1 to s
          print C[i]    //print position p of each student in the order of reg_no
```

Evaluation criteria : **[6 marks]**
Division:

- Read register number and mark of a student and store into the variables - 1 mark
- Find the position p of a student in the arrays A and B - 1 mark
- If position p is vacant, then store the details of the student in the arrays A and B at position p - 1 mark
- If position p is not vacant find next vacant position in the arrays A and B and store it- 1 mark
- Store the position p in an array as per the order of the register number and print the positions after storing m student details - 1 mark
- Correct function name, and number of arguments - 1 mark

display_marks(n, A, B)

```
1.  for i = 1 to n
        do
            if A[i] = -1
                    print -1  //array vacant
            else
                    print A[i]  B[i] separated by a space
```

Evaluation criteria : **[1 mark]**
Division: Print register number and mark of each student in the arrays separated by a space - 1 mark

Design Marks: Total = 3

Set B Part 2

store_details( )

//A: 2D Array of size *n*s* initialised into -1 (to store register number)

//B: 2D Array of size *n*s* initialised into -1 (to store mark)

```
1.  for j = 1 to s
          do
            read  reg_no                      //register number
            read m                            //mark
            compute p = (m+10) mod n          //find position p
            for  i = 1 to n
              do
                    if A[p][i] = -1
                      then
                          A[p][i]  = reg_no
                          B[p][i]   = m
                          break   // break out of  for
```

Evaluation criteria : **[1 mark]**
Division:

- Read the register number and mark of each student and compute position, p - 0.25 mark
- Selection of proper data structure to store the details of the students (if more than one student got the same position p) - 0.75 mark

sort_process()

//A and B are 2-dimensional array of size *n*s* with register number and mark respectively

```
1.  for row = 1 to n
        do
```

```
if  B[row][1]  !=  -1
    do
        //apply any sorting algorithm on B[row], while swapping
        elements in B[row] make changes accordingly in A[row]
        //example given below uses Bubble sort algorithm
        for i = 1 to s
            for  j = 1 to s-i
                do
                    if B[row][j] > B[row][j+1]
                        then
                            swap B[row][j] , B[row][j+1]
                            swap A[row][j] , A[row][j+1]
```

Evaluation criteria : **[1 mark]**
Division:

- Selection of a sorting algorithm - 0.25 mark
- Sort array B (non-decreasing order of mark at position p), and reflect the same changes in array A - 0.75 mark

display_marks()

//each new line prints register number and mark (separated by a comma) of the students that are allotted to position p, where each student is separated by a space

```
1.  for  i =1 to n
            j = 0
            do
                if A[i][j] = -1
                    then
                        print('NULL')
                else
                    while A[i][j] != -1
                        print A[i][j],B[i][j] print('  ')     //prints all the details of
                    the students allotted to same position separated by space
                        j++      //increment j by one
```

Evaluation criteria : **[1 mark]**
Division: Print the details of the students in the given format - **1** mark