GAYATHRI

4AL19CS035

Algorithm

Step1: Start
Step2: void insert (void)
Step3: void delete (void)
Step4: void display (void)
Step5: rear = -1
Step6: front = -1
Step7: Display Enter the size of the queue
Input n
Step8: Display Queue operations using array
Step9: Display 1. ENQUEUE 2. DEQUEUE 3. DISPLAY 4. EXIT
Step10: do
Display Enter your choice
Input choice
switch (choice)
Case 1: insert()
break
Case 2: delete()
break
case 3: display()
break
case 4: EXIT POINT
break

default: Display please Enter a valid choice(1/2/3/4)

. while (choice != 4)
return 0

Step 11: Stop

Void insert()

Step1: Entry

Step2: If (rear == MAX - 1)

Display QUEUE Overflow

Else

if (front == -1)

front = 0

Display Insert the Element in Queue

Input Item

rear = rear + 1

Queue[rear] = Item

Step3: End

Void delete()

Step1: Entry

Step2: if (front == -1 || front > rear)

Display Queue underflow

return

Else

Display Element deleted from Queue is

output Queue[front]

front = front + 1

Step3: End

void display()

Step1 : Entry

Step2 : if (front == -1)

DBplay Queue & Empty

Else

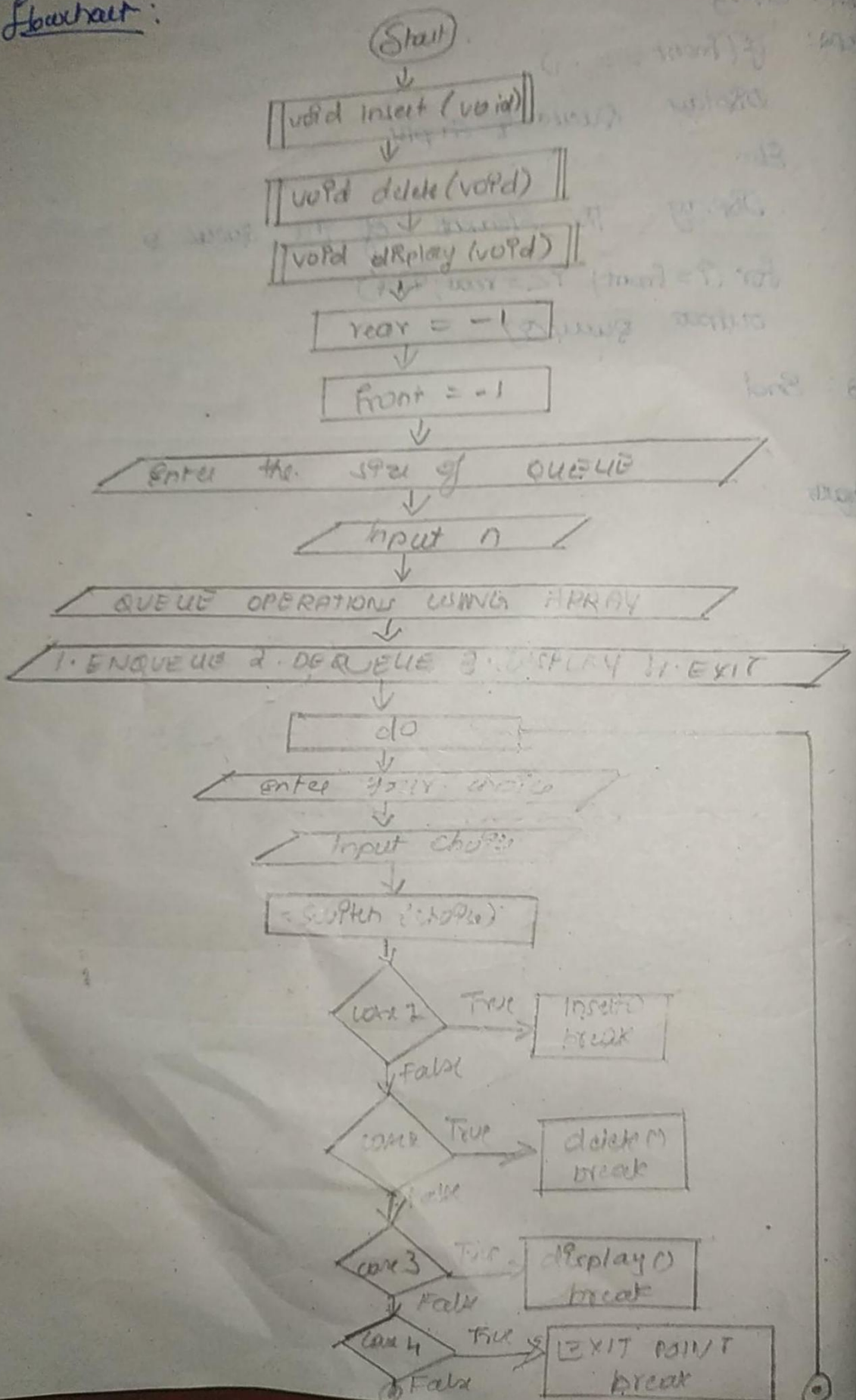
DBplay The Elements of the Queue &

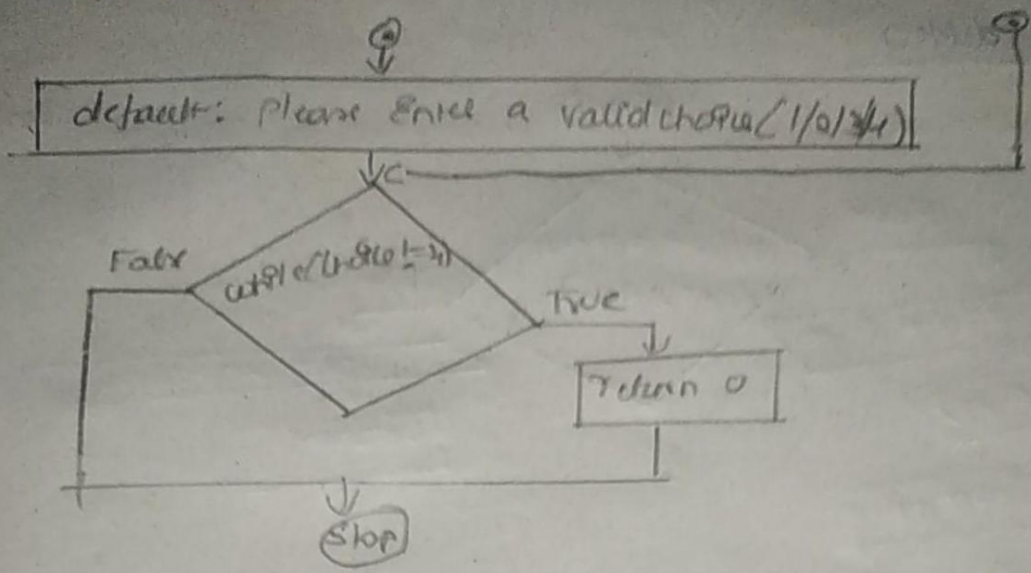
for (9 = front; 9 <= rear; 9++)

output Queue[9]

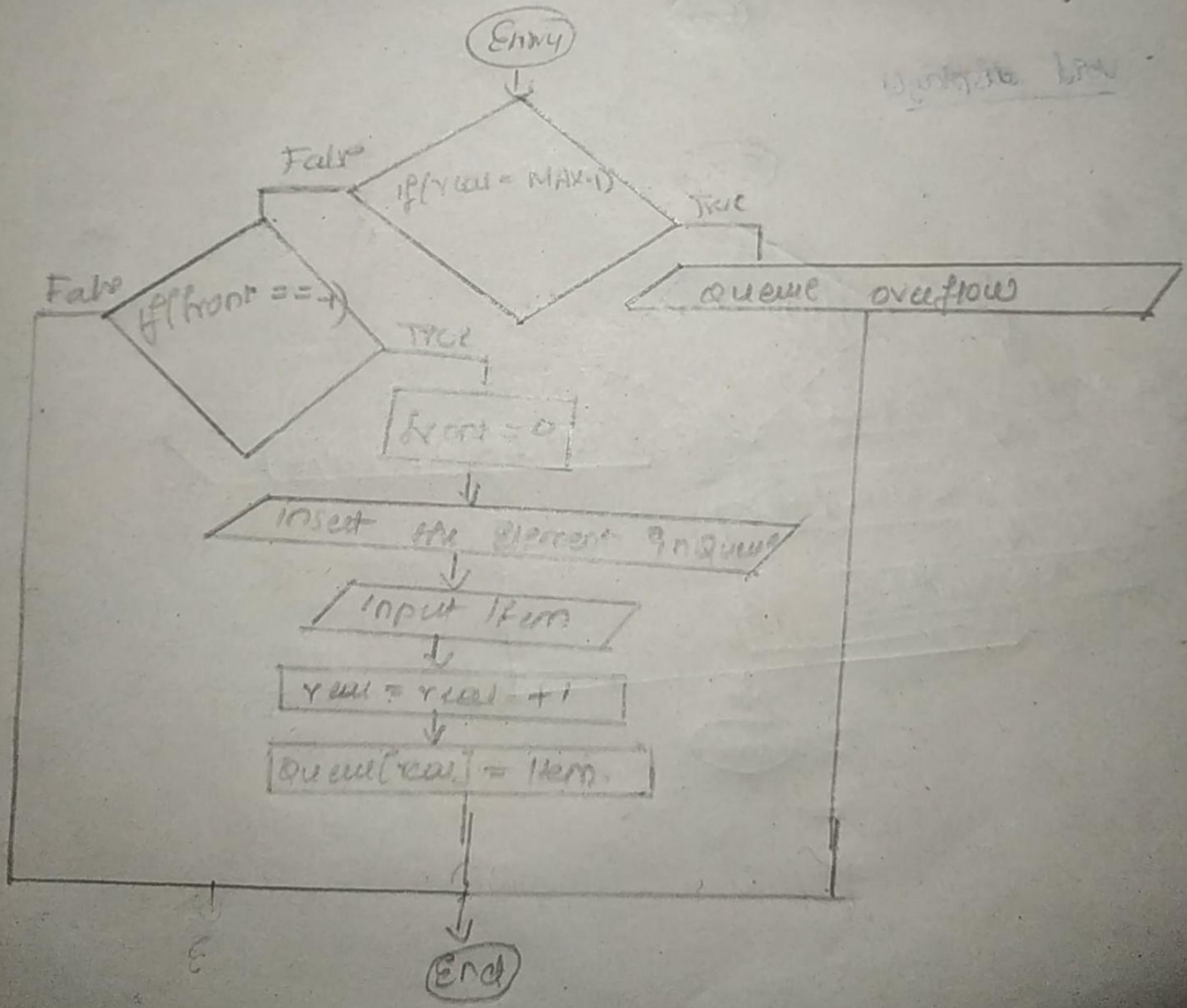
Step3 : End

Flowchart:

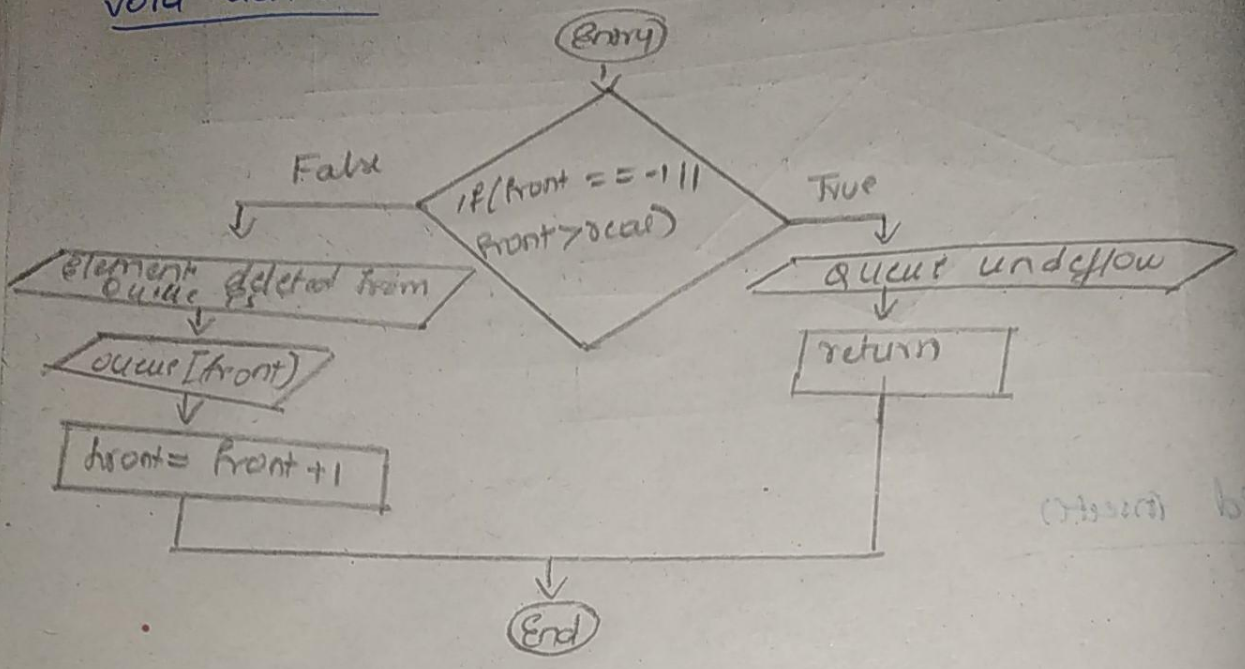
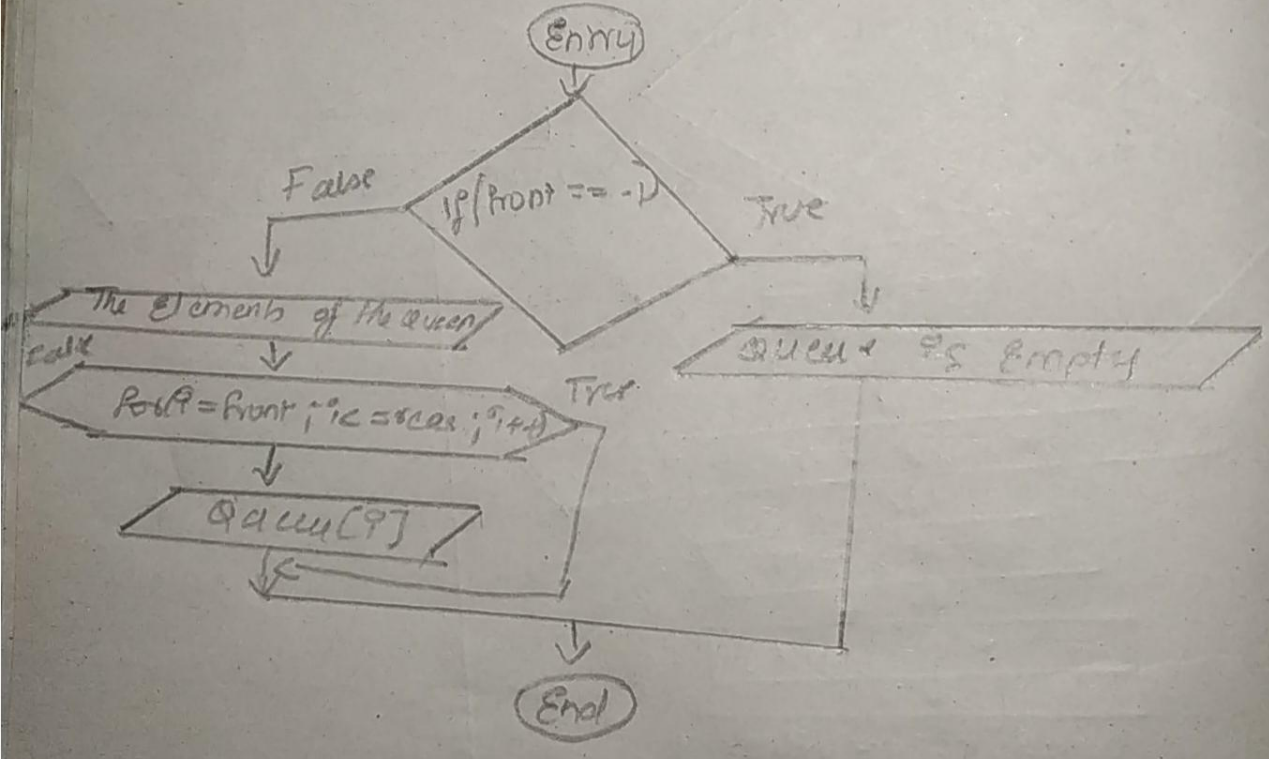

Start
void insert (void)
void delete (void)
void display (void)
rear = -1
front = -1
Enter the size of QUEUE
Input n
QUEUE OPERATIONS USING ARRAY
1. ENQUEUE 2. DEQUEUE 3. DISPLAY 4. EXIT
do
Enter your choice
Input choice
switch (choice)
case 1
True
insert()
break
False
case 2
True
delete()
break
case 3
True
display()
break
False
case 4
True
EXIT POINT
break
False


default: Please Enter a valid choice(1/2/3/4)
C
False
while(choice!=4)
True
Return 0
Stop

void insert()


Enter
False
if(rear = MAX-1)
True
Queue overflow
False
if(front == -1)
True
front = 0
Insert the element in queue
Input Item
rear = rear + 1
Queue[rear] = Item.
End

void delete()



void display()