

Test Cases and Sample output

Case 1 - Leader Replication and DB write : No Coordinator sleeps , No pig dies

```
Displaying the 1d grid
0-unoccupied
1-Pig
2-stone
positions denoted from 0 to 11 with 0 being on the left
[0, 1, 2, 1, 0, 1, 1, 1, 1, 2, 0, 2]
Informing Pigs about the leaders
Iteration : 1
  positions of leader 1 : 5
Pigs leader 1 is incharge of : [0, 4]
  Dead pigs under leader 1 : []
  positions of leader 2 : 3
Pigs leader 2 is incharge of : [3, 5]
  Dead pigs under leader 1 : []
Bird landing position :10
Time taken by the bird :11
The Leader gets the bird time and bird landing positions and sends it to the
affected pig
Pigs to be warned : []
Pigs to be warned : []
  score in the current iteration : 0
Score : 0
0,6,F

4,8,F

3,1,F

5,7,F

y to continue / n to exit : y
```

Conclusion : The pigs are equally distributed between the two pigs . The bird falls in no mans land and hence none of the pigs are affected . The results are written to the database

Case 2 - Logic behind bird attack : No leader is asleep , a pig is killed by the bird (Assuming the same grid as before)

```
Iteration : 3
  positions of leader 1 : 5
Pigs leader 1 is incharge of : [0, 4]
  Dead pigs under leader 1 : []
  positions of leader 2 : 3
Pigs leader 2 is incharge of : [3, 5]
  Dead pigs under leader 1 : []
Bird landing position :8
Time taken by the bird :1
The Leader gets the bird time and bird landing positions and sends it to the
affected pig
Pigs to be warned : [4]
Pigs to be warned : []
Received BIRD APPROACHING message - ID : 4
Received BIRD HIT at (leader):
Pig 4 Killed !
Received status from all Pigs
The Pigs that were hit : [4]
  score in the current iteration : 1
Score : 1
0,6,F

4,-1,T

3,1,F

5,7,F
```

Conclusion : The pigs are equally divided amongst the two coordinators , the bird falls in the pig 4's position . Pig 4 is killed , the result from the database are provided . The DB has a status T indicating its dead and the position is set to -1 to show that it is no longer in the grid .

Case 3 - Fault tolerance , DB read : One pig falls asleep , no pigs die (Assuming the same grid has before)

Leader 2 falling asleep

Iteration : 4

0 6 F

3 1 F

4 -1 T

5 7 F

positions of leader 1 : 5

Pigs leader 1 is incharge of : [0, 3, 5]

Dead pigs under leader 1 : [4]

leader 2 is asleep

Bird landing position :4

Time taken by the bird :1

The Leader gets the bird time and bird landing positions and sends it to the affected pig

Pigs to be warned : []

score in the current iteration : 0

Score : 1

0,6,F

3,1,F

5,7,F

4,-1,T

Conclusion : Since the leader 2 is down , leader 1 reads the pig list from the database and becomes responsible for all the pigs . The bird does not fall in the pigs position and hence no pig dies

Case 4 - Fault tolerance : Leader wakes up (Assuming the same grid has before)

```
[]
Leader 2 waking up.
Iteration : 5
  positions of leader 1 : 5
Pigs leader 1 is incharge of : [3, 5]
  Dead pigs under leader 1 :[]
  positions of leader 2 : 3
Pigs leader 2 is incharge of : [0]
  Dead pigs under leader 1 :[4]
Bird landing position :5
Time taken by the bird :1
The Leader gets the bird time and bird landing positions and sends it to the
affected pig
The bird is falling at the leaders position
Leader takes evasive action and escapes
Pigs to be warned : []
Pigs to be warned : []
  score in the current iteration : 0
Score : 1
0,6,F

4,-1,T

3,1,F

5,7,F
```

Conclusion : Now the leader which was sleeping the previous iteration wakes up , the load has to be balanced again . The two pigs coordinate amongst themselves and share the load. Pigs under leader 1 are 3,5 and pigs under leader 2 are 0 . The leader 2 also keeps track of the dead pig 4 in its local cache.

Case 5 - Consistency : Pig escapes a bird , moves to a new location

```
Displaying the 1d grid
0-unoccupied
1-Pig
2-stone
positions denoted from 0 to 11 with 0 being on the left
[0, 2, 1, 1, 1, 1, 1, 2, 1, 0, 0, 2]
Informing Pigs about the leaders
Iteration : 1
  positions of leader 1 : 3
Pigs leader 1 is incharge of : [5, 4]
  Dead pigs under leader 1 : []
  positions of leader 2 : 6
Pigs leader 2 is incharge of : [0, 3]
  Dead pigs under leader 1 : []
Bird landing position :7
Time taken by the bird :6
The Leader gets the bird time and bird landing positions and sends it to the
affected pig
Pigs to be warned : [5]
Pigs to be warned : []
Received BIRD APPROACHING message - ID : 5
Pig with id 5 with position : 8 takes evasive action moves to : 9
Received status from all Pigs
  score in the current iteration : 0
Score : 0
5,9,F

4,2,F

0,5,F

3,4,F

y to continue / n to exit : y
```

Conclusion : Here the pig with id 5 escapes the pig attack moves to position 9 , this change is immediately propagated to the database and the new location is written to the db (consistency)

Case 6 : Leader gets affected

```
Displaying the 1d grid
0-unoccupied
1-Pig
2-stone
positions denoted from 0 to 15 with 0 being on the left
[1, 2, 0, 1, 1, 0, 2, 1, 0, 1, 2, 1, 2, 1, 1, 0]
Informing Pigs about the leaders
Iteration : 1
  positions of leader 1 : 9
Pigs leader 1 is incharge of : [0, 6, 5]
  Dead pigs under leader 1 : []
  positions of leader 2 : 0
Pigs leader 2 is incharge of : [3, 4, 7]
  Dead pigs under leader 1 : []
bird time ? 0
bird position ? 0
Bird landing position :0
Time taken by the bird :0
The Leader gets the bird time and bird landing positions and sends it to the
affected pig
Pigs to be warned : []
The bird is falling at the leaders position
Leader takes evasive action and escapes
Pigs to be warned : []
  score in the current iteration : 0
Score : 0
0,11,F

6,7,F

5,14,F

3,4,F

4,3,F

7,13,F

y to continue / n to exit :
```

Conclusion : Here the leader gets affected . Since the leader knows its position it escapes the attack