Algorithm 1 Animal Barn (B, n, m)

Input: B-an array of strings representing $n+m$ animals in a barn (chicken = "chicken", rabbit = "rabbit"), n-number of chickens from B, m-number of rabbits from B
Output: "chicken", if the last element of B is chicken, otherwise, "rabbit"

```
initChickenN = n
i = 1
while B.size > 1 do
    first ← delete a random element in B and assign
    second ← delete a random element in B and assign
    if first = second = "chicken" then
        B.insert("rabbit")
        n− = 2
        m+ = 1
    else if first = second = "rabbit" then
        B.insert("rabbit")
        m− = 1
    else //the program picked "rabbit" and "chicken"
        B.insert("chicken")
        m− = 1
    end if
    i++
end while
return B
```

Loop invariant : $initChickenN \ \% \ 2 = n \ \% \ 2$

Initialization : Before the loop begins, $initChickenN = n$, so $initChickenN \ \% \ 2 = n \ \% \ 2$ holds true.

Maintenance : Assume that the loop invariant holds at the beginning of k'th iteration. Then it must mean that if $initChickenN$ was even, then n also has to be even, else both $initChickenN$ and n are odd at this point.

In the loop body, there are three cases, where 1) if the taken out animals are both chickens, n is decrease by 2 and m is increased by 1. 2) else if they are both rabbits then m is decreased by 1. 3) chicken rabbit so m is decreased by 1.

The only condition that changes the number of n is 1), and whether n was even number or odd number, it would not change the property of n being odd or even by decreasing it by 2. And Thus in this case the loop invariant holds again at the beginning of the next loop.

Termination: Loop terminates when $B.size = 1$ as the size of B decreases by 1 for each

iteration. At that point, there is only one animal left in a barn (B). If initial n value (value of