Q1

a)
```
if the cheese is in sight:
    detect the location of the cheese
    if the cheese is in a horizontal or vertical direction:
        move in the direction until reaching the cheese
    else:
        generate a coordinate of the cheese relative to the mouse (X, Y) (X, Y ∈ {E, S, W, N})
        randomly select one direction in the relative coordinate
        move one slot in the selected direction
        move one slot in another direction
else if the mouse is next to the boundary of the plane:
    find directions without boundaries
    randomly select one of them and move one slot
else:
    randomly select one direction from E, S, W, N and move one slot
```

b)
```
set CheeseNotFound = True
set DirectionList = [E, NE, S, SE, W, NW, N, SW]
while cheeseNotFound:
    for i in range(1, 6):
        if i % 2 = 1:
            detect the nearest slot in direction DirectionList[i]
            if the cheese exists in the slot:
                CheeseNotFound = False
            else:
                detect the second nearest slot in direction DirectionList[i]
                    if the cheese exists in the slot:
                        CheeseNotFound = False
                    else:
                        CheeseNotFound = True
        else if i % 2 = 0:
            detect the nearest slot in the direction[i]
            if the cheese exists in the slot:
                CheeseNotFound = False
            else:
                CheeseNotFound = True
return CheeseNotFound
```

c)
```
set A = findCheese()
while A = True, repeat:
    move()
    set A = findCheese()
 until A = False
```

Q2

a) pseudocode of function LengthConvert(A, m, n):

```
examine the first digit of A
if the first digit of A is 0:
    insert (n – m) 0s between the first digit and the second digit of A
else:
    insert (n – m) 1s between the first digit and the second digit of A
return A of length n
```

b)

```
compare m with n
set r to be equal to the bigger one in m and n
set A' = LengthConvert(A, m, r)
set B' = LengthConvert(B, n, r)
give A' a new representation: A'(r)A'(r-1)A'(r-2)…A'(2)A'(1)
give B' a new representation: B'(r)B'(r-1)B'(r-2)…B'(2)B'(1)

give examples of binary addition:
    0 + 0 = 0
    0 + 1 = 1
    1 + 0 = 1
    1 + 1 = 10

pseudocode of function AddTwo(x, y):
    look up the examples
    let s be the entry
    set s* to be the first digit of s
    set s** to be the second digit of s
    return (s*, s**)

pseudocode of function AddThree(x, y, z):
    set (C, D) = AddTwo(x, y)
    set (E, F) = AddTwo(D, z)
    set (G, H) = AddTwo(C, E)
    return (H, F)

set z(1) = 0
for i in range(1, r):
    (p, q) = AddThree(A'(i), B'(i), z(i))
    set z(i+1) = p
    set t(i) = q

return the number z(r+1)t(r)t(r-1)…t(2)t(1)
```