

## 6. Το πρώτο παιχνίδι

Ένα πρώτο βήμα για να γίνει η εφαρμογή που κατασκευάσατε στην επέκταση 5.4 παιχνίδι, είναι να κινείται το ποντίκι με τα πλήκτρα του πληκτρολογίου. Η εντολή που το επιτρέπει αυτό είναι η **Greenfoot.isKeyDown()**. Η φράση **isKeyDown** σημαίνει «Είναι το πλήκτρο πατημένο» χωρίς να αναφέρεται σε κάποιο πλήκτρο συγκεκριμένα. Το πλήκτρο δηλώνεται μέσα στην παρένθεση και μπορεί να είναι οποιοδήποτε πλήκτρο του πληκτρολογίου. Για να χρησιμοποιήσουμε τα βελάκια μέσα στην παρένθεση θα βάλουμε τις λέξεις ("up","down","left","right"). Έτσι αντικαθιστούμε τη μέθοδο **randomMove** με τη μέθοδο **checkKeyboard** μέσα στην **act** της κλάσης **mouse**.

### Κώδικας 6.1:

```
public void act()
{
    checkKeyboard();
    touchingCheese();
}
void checkKeyboard() {
    if (Greenfoot.isKeyDown("left"))
    {
        turn(-90);
    }
    if (Greenfoot.isKeyDown("right"))
    {
        turn(90);
    }
    if (Greenfoot.isKeyDown("up")) {
        move(1);
    }
}
```

Η μέθοδος  
**touchingCheese**  
δεν αλλάζει.

Παρατηρείτε ότι με το αριστερό και το δεξί πλήκτρο το ποντίκι στρίβει, ενώ με το πάνω προχωράει προς την κατεύθυνση που έχει τη δεδομένη χρονική στιγμή.

Μπορούμε να σταματήσουμε το παιχνίδι όταν το αντικείμενο **lemur** ακουμπήσει το ποντίκι. Αυτό μπορεί να γίνει με την εντολή **Greenfoot.stop()**. Η οποία θα πρέπει να εκτελεστεί όταν το **lemur** ακουμπήσει το **mouse**, και κατά συνέπεια πρέπει να μπει στον κώδικα του **lemur**.

### Κώδικας 6.2:

```
void touchingMouse() {
    if (isTouching(mouse.class))
    {
        removeTouching(mouse.class);
        Greenfoot.stop();
    }
}
```

Το παιχνίδι θα πρέπει να σταματάει και στην περίπτωση που το ποντίκι φάει όλα τα τυράκια. Με κάποιο τρόπο θα πρέπει να ξέρουμε πόσα τυράκια έχουν φαγωθεί (το σκορ). Δεν είναι ανάγκη όλες οι μορφές να ξέρουν το σκορ, αρκεί να το υπολογίζει το αντικείμενο

της κλάσης `mouse`. Αρχικά το σκορ είναι 0 καθώς το ποντίκι τρώει τυράκια, το σκορ μπορεί να αυξάνεται. Τέλος το σκορ είναι ακέραιος αριθμός (δεν έχει δεκαδικά ψηφία). Έτσι δημιουργούμε μια μεταβλητή ακεραίου με όνομα `score` και αρχική τιμή 0 (εντολή `int score = 0`). Την εντολή αυτή τη βάζουμε πάνω από τις συναρτήσεις ώστε όλες οι συναρτήσεις να έχουν πρόσβαση στη μεταβλητή `score`.

Κάθε φορά που το ποντίκι τρώει ένα τυράκι η τιμή `score` πρέπει να αυξηθεί. Έτσι στην τιμή `score` προσθέτουμε το 1 (`score + 1`). Αυτή πρέπει να είναι η νέα τιμή της μεταβλητής `score` (`score = score + 1`). Η διαδικασία αυτή γίνεται όταν εξαφανίζουμε ένα τυράκι.

#### Κώδικας 6.3:

```
int score = 0;
public void act()
{
    checkKeyboard();
    touchingCheese();
}
void touchingCheese() {
    if (isTouching(chesse.class))
    {
        removeTouching(chesse.class);
        score = score + 1;
        System.out.println(score);
    }
}
```

Η μέθοδος `checkKeyboard` δεν αλλάζει.

Κάθε φορά που το ποντίκι «τρώει» ένα τυράκι, εμφανίζεται ένα παράθυρο με το σκορ. Ωστόσο, δεν είναι αυτή η λειτουργικότητα που επιθυμούμε. Μπορούμε να κάνουμε το παιχνίδι να τελειώσει όταν το `score` γίνει 3. Χρησιμοποιούμε την εντολή `Greenfoot.stop()` και ελέγχουμε αν η τιμή της μεταβλητής `score` είναι 3 χρησιμοποιώντας τον τελεστή `==`.

#### Κώδικας 6.4:

```
void touchingCheese() {
    if (isTouching(chesse.class))
    {
        removeTouching(chesse.class);
        score = score + 1;
        if (score == 3)
        {
            Greenfoot.stop();
        }
    }
}
```

#### Επέκταση 6.1:



Μπορείτε να εισάγετε κατάλληλες εντολές ώστε το ποντίκι να κάνει μεταβολή όταν πατηθεί το κάτω πλήκτρο του πληκτρολογίου.

#### Επέκταση 6.2:



Τι θα κάνετε για να βάλετε περισσότερα τυράκια στο παιχνίδι; Τι θα πρέπει να αλλάξετε στον κώδικα της κλάσης `mouse`;