



Snake!

林柏佑 Lin, Po-Yu

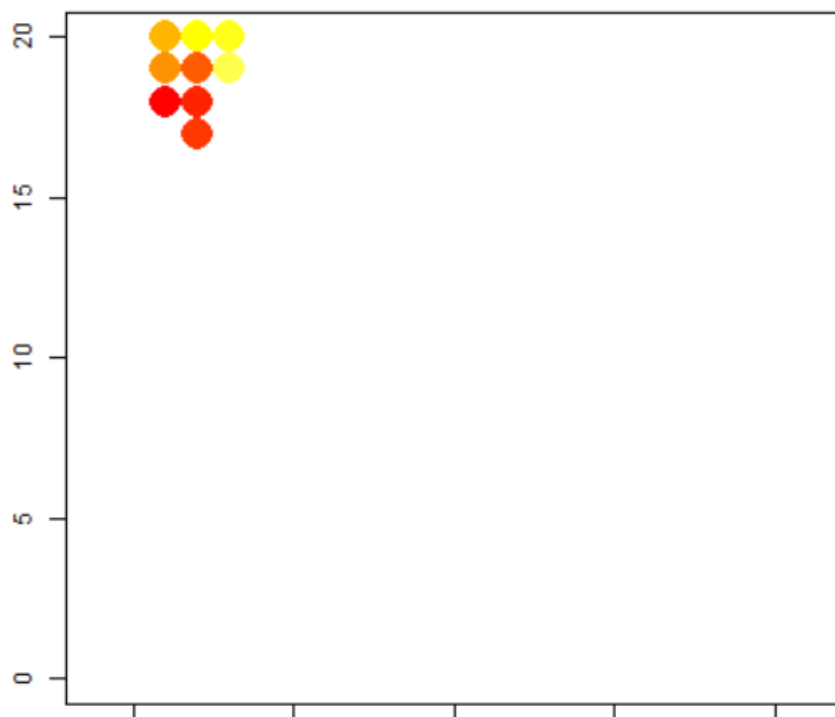
Institution of Ecology and Evolutionary Biology

The second year of a master

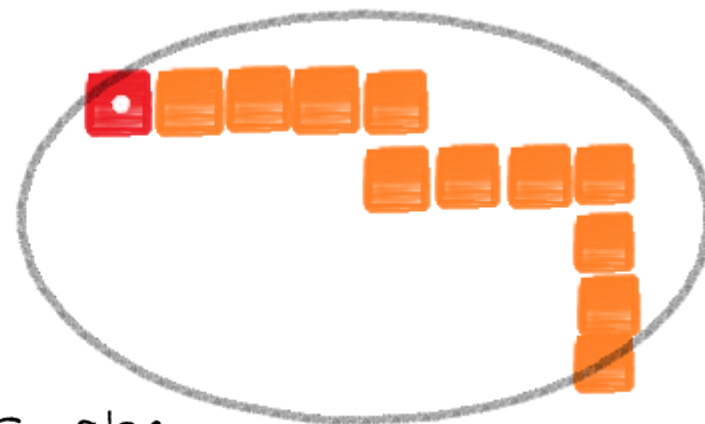


Animation of moving snake

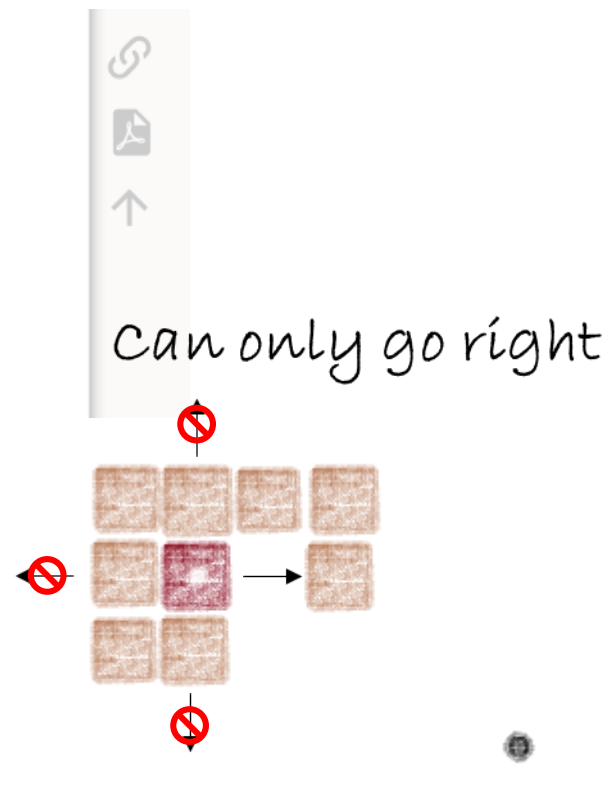
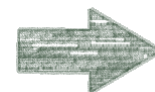
Try to reproduce this:

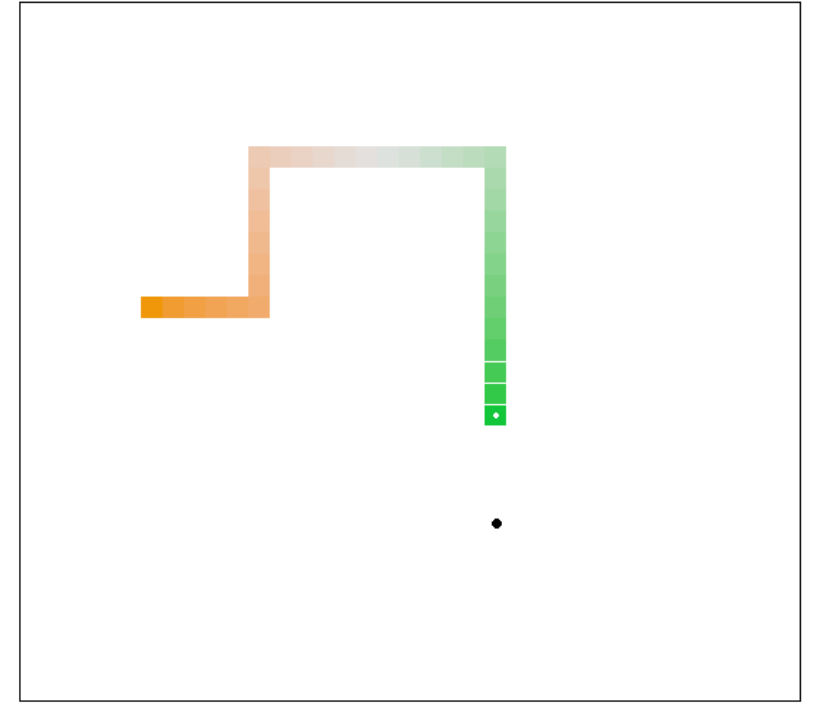
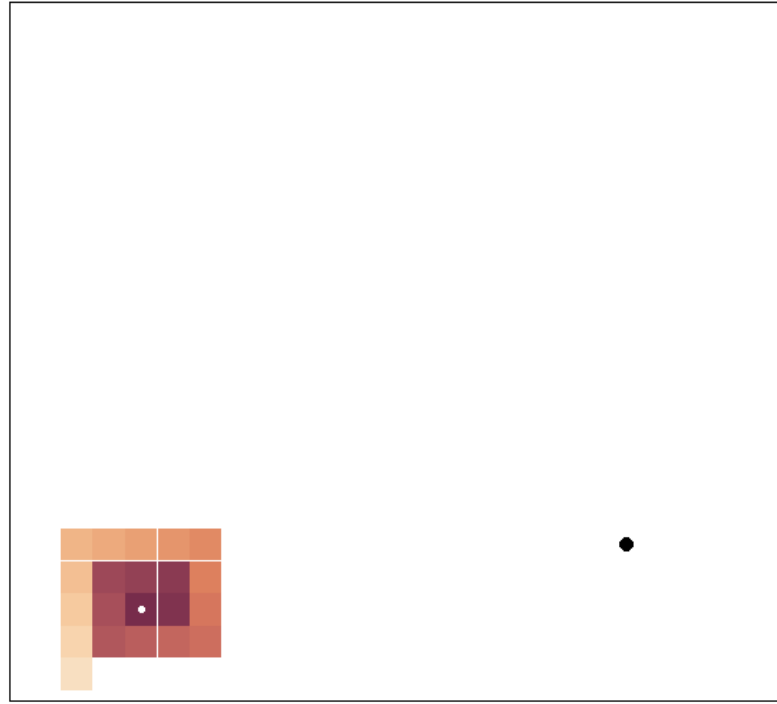
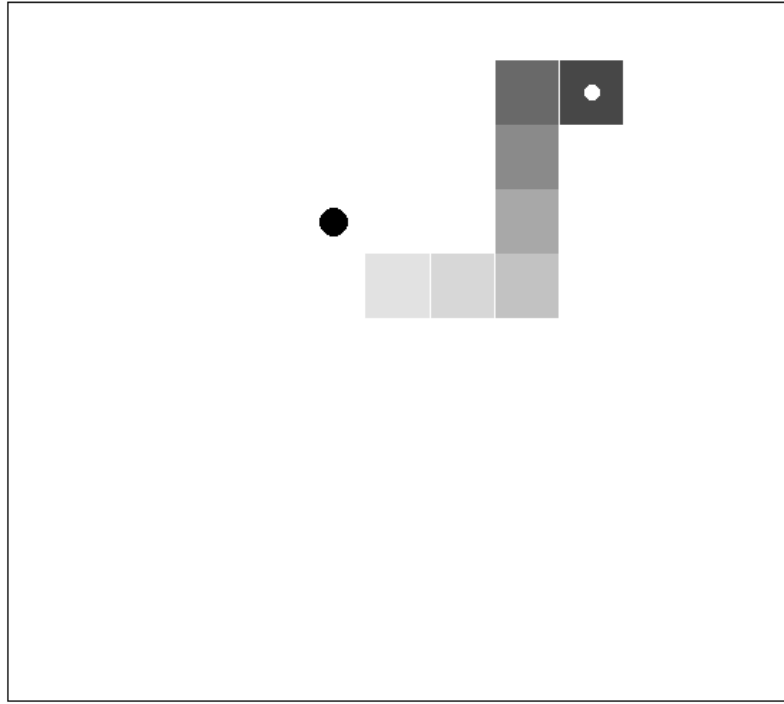


Food



Snake





A function:
`animated.snake()`

The arguments:
color, moving mode, speed, size of playground

R code

```
2 animated.snake <- function(snake.length = 3, square.size = 20, no.step = 0, speed = 300, mode = "shortest",  
  color = "heat"){
```

```

23 # initiate -----
24 # ----
25 snake <- matrix(NA, ncol = 2, nrow = square.size)
26 snake[1,] <- c(round(square.size/2), 0)
27 for (i in 2:snake.length){
28   snake[i,] <- snake[i-1,] + c(0, 1)
29 }
30 body <- paste(snake[,1], snake[,2], sep = ",")
31 fd <- outer(0:square.size, 0:square.size, FUN = function(x, y){
32   food <- unlist(strsplit(sample(letters, 1), ""))
33   food <- as.numeric(c(food[1], food[2]))
34   eat.food <- FALSE
35 }
36 snake_food <- rbind(food, snake)
37 # random ----
38 if (mode == "random"){
39   # boundary & body
40   # body <- str_c(snake[-1,1], snake[-1,2], sep = ",")
41   if ((snake[2,2] >= square.size) | (snake[2,2] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
42     nt[rownames(nt) != "up",] # up
43     if ((snake[2,2] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
44       nt["down",] # down
45       if ((snake[2,1] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
46         nt["left",] # left
47         if ((snake[2,1] >= square.size) | (snake[2,1] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
48           nt[rownames(nt) != "right",] # right
49           # no way
50           if (is.null(nrow(nt))) {title(main = "No way out"); break}
51           snake[1,] <- snake[2,] + nt[1,]
52         }
53       }
54     }
55   }
56 }
57 # shortest ----
58 else if (mode == "shortest"){
59   if (food[1] > snake[2,1]){
60     if ((!paste(snake[2,1]+1, snake[2,2], sep = ",") %in% body)) {
61       nt["right",] # can go right
62     } else if (food[2] <= snake[2,2]){
63       if ((!paste(snake[2,1], snake[2,2]+1, sep = ",") %in% body)) {
64         nt["down",] # can go down
65       } else if ((!paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
66         nt["up",] # can't go right and up
67       } else if ((!paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
68         nt["left",] # can't go right, down and left
69       } else {title(main = "No way out"); break}
70     }
71   } else{
72     # can't go right, food is to the left
73     if ((!paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
74       nt["up",] # can go up
75     } else if ((!paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
76       nt["down",] # can't go right and up
77     } else if ((!paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
78       nt["left",] # can't go right, up and left
79     } else {title(main = "No way out"); break}
80   }
81 }
82 else if (food[1] < snake[2,1]{
83   if (food[2] < snake[2,2]){
84     nt["up",] # can go up
85   } else{
86     # can't go up, food is below
87     if ((!paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
88       snake[1,] <- snake[2,] + nt[1,]
89     }
90   }
91 }
92 }
93 # clockwise ----
94 else if (mode == "clockwise"){
95   clock_count <- 0
96   while (TRUE){
97     # just food
98     if (clock_count == 1){
99       if (clock_dir == "left"){
100         # check left
101         if ((snake[2,1] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
102           nt["up",] # not go up
103           else {nt <- nt[clock_dir,]; break}
104         }
105       } else if (clock_dir == "up"){
106         # check up
107         if ((snake[2,2] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
108           {clock_dir <- "right"; clock_count <- clock_count + 1}
109           else {nt <- nt[clock_dir,]; break}
110         }
111       } else if (clock_dir == "right"){
112         # check right
113         if ((snake[2,1] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
114           {clock_dir <- "down"; clock_count <- clock_count + 1}
115           else {nt <- nt[clock_dir,]; break}
116         }
117       } else if (clock_dir == "down"){
118         # check down
119         if ((snake[2,2] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
120           "left"; clock_count <- clock_count + 1} # not go left
121           else {nt <- nt[clock_dir,]; break}
122         }
123       }
124     }
125     else if (clock_count == 2){
126       # just food
127       if (clock_dir == "left"){
128         # check left
129         if ((snake[2,1] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
130           nt["up",] # not go up
131           else {nt <- nt[clock_dir,]; break}
132         }
133       } else if (clock_dir == "up"){
134         # check up
135         if ((snake[2,2] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
136           {clock_dir <- "right"; clock_count <- clock_count + 1}
137           else {nt <- nt[clock_dir,]; break}
138         }
139       } else if (clock_dir == "right"){
140         # check right
141         if ((snake[2,1] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
142           {clock_dir <- "down"; clock_count <- clock_count + 1}
143           else {nt <- nt[clock_dir,]; break}
144         }
145       } else if (clock_dir == "down"){
146         # check down
147         if ((snake[2,2] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
148           "left"; clock_count <- clock_count + 1} # not go left
149           else {nt <- nt[clock_dir,]; break}
150         }
151       }
152     }
153     else if (clock_count == 3){
154       # just food
155       if (clock_dir == "left"){
156         # check left
157         if ((snake[2,1] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
158           nt["up",] # not go up
159           else {nt <- nt[clock_dir,]; break}
160         }
161       } else if (clock_dir == "up"){
162         # check up
163         if ((snake[2,2] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
164           {clock_dir <- "right"; clock_count <- clock_count + 1}
165           else {nt <- nt[clock_dir,]; break}
166         }
167       } else if (clock_dir == "right"){
168         # check right
169         if ((snake[2,1] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
170           {clock_dir <- "down"; clock_count <- clock_count + 1}
171           else {nt <- nt[clock_dir,]; break}
172         }
173       } else if (clock_dir == "down"){
174         # check down
175         if ((snake[2,2] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
176           "left"; clock_count <- clock_count + 1} # not go left
177           else {nt <- nt[clock_dir,]; break}
178         }
179       }
180     }
181     else if (clock_count == 4){
182       # just food
183       if (clock_dir == "left"){
184         # check left
185         if ((snake[2,1] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
186           nt["up",] # not go up
187           else {nt <- nt[clock_dir,]; break}
188         }
189       } else if (clock_dir == "up"){
190         # check up
191         if ((snake[2,2] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
192           {clock_dir <- "right"; clock_count <- clock_count + 1}
193           else {nt <- nt[clock_dir,]; break}
194         }
195       } else if (clock_dir == "right"){
196         # check right
197         if ((snake[2,1] >= square.size) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
198           {clock_dir <- "down"; clock_count <- clock_count + 1}
199           else {nt <- nt[clock_dir,]; break}
200         }
201       } else if (clock_dir == "down"){
202         # check down
203         if ((snake[2,2] <= 0) | (paste(snake[2,1], snake[2,2], sep = ",") %in% body)) {
204           "left"; clock_count <- clock_count + 1} # not go left
205           else {nt <- nt[clock_dir,]; break}
206         }
207       }
208     }
209   }
210 }

```