```
(* Read command line arguments *)
if Array.length Sys.argv <> 4
then print_string "USAGE: ./langston <#rounds> <input_file> <output_file>\n"
else ()

let rounds = ref (int_of_string Sys.argv.(1))
let inFile = ref (Sys.argv.(2))
let outFile = ref (Sys.argv.(3))

(*Function for reading input file line by line*)
let rec read_file input_channel lines =
        try
                read_file input_channel ((input_line input_channel) :: lines)
        with e ->
                lines;;


let row = ref 0
let col = ref 0
let dir = ref 'A'
let k = ref 0

(*Parse input*)
let get_position str =
        let rec get_length str idx =
                if str.[idx] < '0' || str.[idx] > '9'
                then 0
                else 1 + get_length str (idx+1)
        in
                let row_len = get_length str 1 in
                row := int_of_string (String.sub str 1 row_len);
                let col_len = get_length str (2 + row_len) in
                col := int_of_string (String.sub str (2+row_len) col_len);
                dir := str.[4 + col_len + row_len]

let parse_input lines =
        get_position (List.hd lines);
        k := List.length lines - 1;
        Array.of_list (List.rev (List.tl lines))

let board = parse_input (read_file (open_in !inFile) []);;


(*start simulating the langston's ants*)
let move () =
        if board.(!row).[!col] = '0'
        then
                (*move right*)
                (board.(!row).[!col] <- 'X';
                if !dir = 'A'
                then
                        (dir := '>';        col := (!col + 1) mod !k)
                else if !dir = '>'
                then
                        (dir := 'V';        row := (!row + 1) mod !k)
                else if !dir = 'V'
                then
                        (dir := '<';        col := (!col + !k - 1) mod !k)
                else
                        (dir := 'A';        row := (!row + !k - 1) mod !k))
        else
                (*move left*)
                (board.(!row).[!col] <- 'O';
                if !dir = 'V'
                then
                        (dir := '>';        col := (!col + 1) mod !k)
                else if !dir = '<'
                then
                        (dir := 'V';        row := (!row + 1) mod !k)
                else if !dir = 'A'
                then
                        (dir := '<';        col := (!col + !k - 1) mod !k)
                else
                        (dir := 'A';        row := (!row + !k - 1) mod !k))
;;

for i = 1 to !rounds do
        move();
done;;


(*print output in the required file*)
open Printf;;

let out_chan = open_out !outFile;;
Array.iter (fprintf out_chan "%s\n") board;;
fprintf out_chan "(%d,%d) %c" !row !col !dir;;
close_out out_chan;;
```