

Exercise 03 - Langton's Ant (`ant.c`)

About this exercise

In this exercise we want to simulate Langton's ant. Langton's ant moves on a board divided in colored squares (similar to a chessboard). In our case this surface consists of 10x10 squares. Every square on this field is initially colored white (represented by a single dot `.`). Langton's ant is located on a square on this board and it is facing west (left).

One step of the ant consists of the following procedure: First, the ant checks the color of the square on which it currently sits. If the square is white (`.`) it turns right (i.e., if it was facing west/left, it is now facing north/up). It then leaves its pheromones by flipping the color of the square, i.e., the square's color changes to black (represented by `#`). It then advances one step into the current direction.

If the square is black (`#`), it turns left (i.e., if it was facing south, it is now facing east/right), flips the color of the square (which hence becomes white `.`) and advances by one step.

If the ant leaves the board it dies.

Write a program that reads how many steps the ant should make, then reads an initial position (first x - horizontal, then y - vertical position) where in the best tradition of array indices 0 represents the left/topmost coordinate.

Your program should show the state of the surface after any one of the following conditions is satisfied:

- Either the ant has completed the entered number of steps, or
- It fell off the board and died.

If the ant is still on the board and hence alive its position should be marked by `'a'` if it is on a white square or `'A'` if it is on a black square. If the ant died because it fell off the board before completing all steps there is no ant on the board.

You find a short animation of the ant at Wikipedia. This animation corresponds to the ant in our program with the starting position 4 4.

In your solution you should store the position and the direction of the ant in a `struct`. For the direction you can pick any data structure.

Examples

```
$ ./ant
Enter number of turns: 0
Enter start position: 4 4
```

```

.....
.....
.....
.....
....a.....
.....
.....
.....
.....
.....

$ ./ant
Enter number of turns: 1
Enter start position: 4 4
.....
.....
.....
....a.....
....#.....
.....
.....
.....
.....
.....

$ ./ant
Enter number of turns: 52
Enter start position: 3 4
.....
.....
.....
...#..#...
..#a##.#..
..#....#..
...#..#...
....##....
.....
.....

$ ./ant
Enter number of turns: 4
Enter start position: 6 3
.....
.....
.....##..
.....A#..
.....

```

```

.....
.....
.....
.....
.....

$ ./ant
Enter number of turns: 250
Enter start position: 4 4
.....##...
..#####..
.#.#.#.#.#
#..#.#..#.#
#..##.#.#.#
#..#.#.#.#
#..#.#.#.#
#..###...#
.#####.#
..##.#..#
.....##.

```