SNAKE GAME IN C

ABHAY KOUSHAL
111903133

INTRODUCTION

- I've developed the game called 'snake' in C language .
- This game has been around since the earliest days of home computing and has re-emerged in recent years on mobile phones.
- It isn't the world's greatest game, but it does give you an idea of what you can achieve with a relatively simple C program, and perhaps the basis by which to extend the principles and create more interesting games of your own.
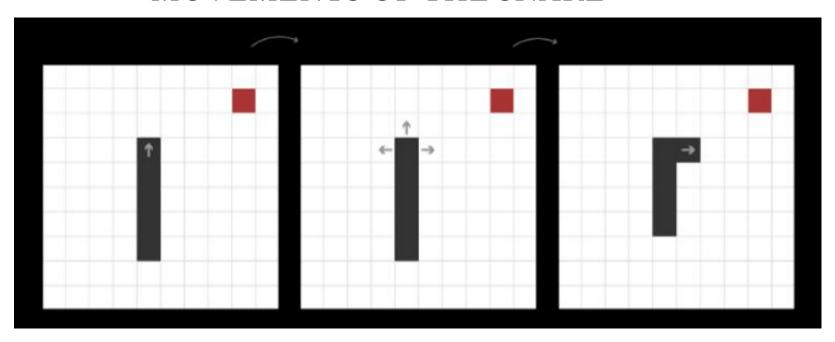
RULES

Snake has simple rules:

1. The world is a grid.
2. The snake can only travel orthogonally along this grid.
3. This world has a border that kills the snake on contact in medium and hard mode.
4. The snake cannot stop moving.
5. If the snake runs into itself, it dies.
6. Every time the snake eats, it grows longer.
7. The goal is to grow as long as possible.

IDEA OF PLAYING THE GAME

When playing the game, there is a decision to make each time the snake takes a step forward: continue straight, turn left, or turn right.

First assessing the state of the world that the snake lives in, then choosing the move that will keep it alive and continue to grow longer.
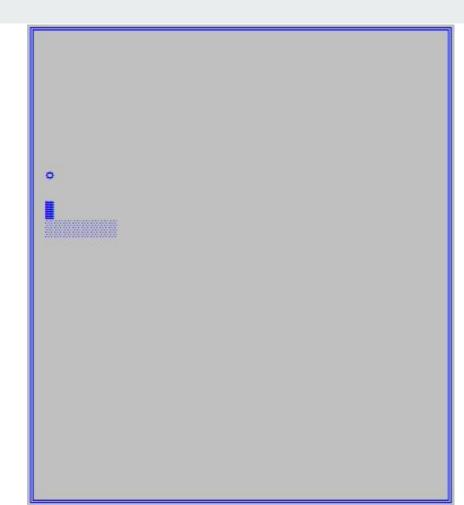
MOVEMENTS OF THE SNAKE

GAMEPLAY

Different modes in the game:

1. BEGINNER
2. EASY
3. MEDIUM
4. HARD

1.Beginner

- <u>Field is open:</u> snake can go from the boundary and can emerge from the other side.
- <u>Scoring is easy:</u> on eating one frog the score increases by 10.
- <u>Movement of snake is not that fast:</u> on eating a frog, the speed increased is only a little.
  - Hence, the user finds it a lot easier to control the snake.
  - Chances for the snake to die are quite less.

2.Easy

- <u>Field is open:</u> snake can go from the boundary and can emerge from the other side.
- <u>Scoring is not that easy:</u> on eating one frog the score increases by 5.
- <u>Movement of snake is bit fast:</u> on eating a frog, the speed increases.
  - Hence, the user finds it a bit easier to control the snake.
  - Chances for the snake to die are quite less but are more than the beginner mode.

3.Medium

- <u>Field is closed:</u> game gets over on touching the boundary.
- <u>Scoring is bit difficult:</u> on eating one frog the score increases by 5.
- <u>Movement of snake is bit fast:</u> on eating a frog, the speed increases.
  - The user faces bit difficulty.
  - Chances for the snake to die are moderate.

4.Hard

- <u>Field is closed:</u> game gets over on touching the boundary.
- <u>Scoring is difficult:</u> on eating one frog the score increases only by 3.
- <u>Movement of snake is fast:</u> on eating a frog, the speed increases.
  - This is the most difficult level for the player.
  - Chances for the snake to die are highest.

METHODOLOGY

We will create different function for each functionality in the game such as for creating boundary/wall, snake movement and many more.

Learning Some Basic Syntax for Developing GUI.
We can add colors and we can print where you want to.

You need to learn basic functionalities like: -
1) gotoxy(x,y)
2) ANSI TERMINAL COLORS
3) random() and srand()
4) kbhit()

Step 1: Welcome Page of the Game

Using above select proper background and font, welcome the user by showing game title.

Step 2: A Loading Screen.

Step 3: Play the game or view highscore.

Step 4: Selecting the Level and mode of Game.

After displaying the game title, show the levels and ask user to select the level. Save the level selected by user in variable and use it to increase the difficulty level of user.

ALGORITHMS

- **Food creation:**
  - In this, we will use predefined function to create a random number inside the boundary.
  - Each time a function is called it generates a random number.
  - In C, there is a random function srand( ).
  - We used it to place the food at any point on the screen.
- **Changing Direction:**
  - The direction of snake can be changed using kbhit() function .
  - When you press the character accordingly it will change the direction of the snake.
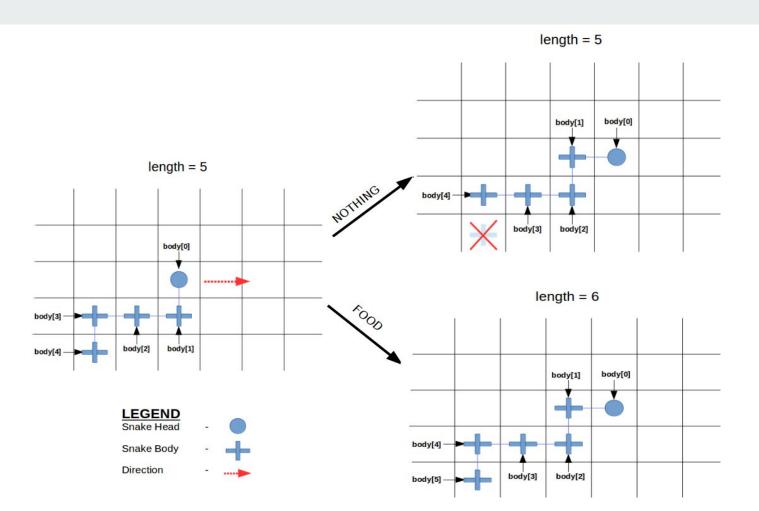
- **Increasing size of snake:**
  - When snake moves forward its size increases (when the coordinate of snake match with food coordinate then the size of snake is increased).
  - We achieve this by delete the tail and increment the head.

- **Increasing score:**
  - When snake gets the food, the score of the user is also increased.

- **Increasing speed:**
  - When snake gets the food, the speed of the snake is also increased.

length = 5

length = 5

NOTHING

length = 6

FOOD

body[0]

body[1]

body[0]

body[4]

body[3]

body[2]

body[1]

body[0]

body[3]

body[2]

body[1]

body[4]

body[4]

body[3]

body[2]

body[5]

LEGEND

Snake Head   -

Snake Body   -

Direction   -

- **Game over Condition:**
  - When snake touch the boundary of the field, the game will be over.
  - Also when the snake touches itself, the game gets over.
  - This can be done by comparing the coordinate of boundary with the snake coordinate and by comparing the coordinate of the snake with itself.

- **Quit Page:**
  - At last step, after game is over, display the user's score and check whether it is greater than the previous highscore.
  - Then ask user if he wants to continue playing or not.