

## **Connect four**

is a two-player connection game in which the players first choose a character (@-0) to play with and then take turns dropping in different places from the top into a seven-column, six-row vertically suspended grid. The pieces fall straight down, occupying the next available space within the column. The objective of the game is to be the first to form a horizontal or vertical line of four of one's own characters (discs).

First , the first player is prompted to enter the number of column to drop his discs in (0) , then the second player is prompted to enter the number of column to drop his discs in(@) and so, on .The player cannot drop in a full column . The game is finished when one of the players connects for discs horizontally or vertically. The game is also finished when there is a draw.

## **Data Structure Used**

The data structure used here is a class **LINKED LIST** having the following members including getters , setters and constructors :-

Member type	Description
Char item	For holding the slot's character
linked_list *next	Pointing to the next slot (down )
linked_list *pre	Pointing to the previous slot (up)
linked_list *last	Pointing to the last slot (top)
linked_list *r	Pointing to the right slot
linked_list *l	Pointing to the lift slot
void add (char i);	Adding items to the list
bool isEmpty();	Checking if the list is empty

## Functions

Prototype	Description
void display();	Display the board on screen
void printLine () ;	Draw the board
void printSides();	Draw the board

<code>void connect();</code>	Connects the slots horizontally
<code>bool check (int col);</code>	Checking if there is a winner
<code>bool checkV (linked_list *col);</code>	Checking if there is 4 identical vertical slots
<code>bool checkH (linked_list *col );</code>	Checking if there is 4 identical horizontal slots
<code>linked_list *fnd (int i);</code>	Find Column # i

## Global Variables

Type	Description
<code>linked_list c1(1);</code>	Column #1
<code>linked_list c2(2);</code>	Column #2
<code>linked_list c3(3);</code>	Column #3
<code>linked_list c4(4);</code>	Column #4
<code>linked_list c5(5);</code>	Column #5
<code>linked_list c6(6);</code>	Column #6
<code>linked_list c7(7);</code>	Column #7

# TEST CASES Horizontal Win

1	2	3	4	5	6	7

player 1 drop where?

1	2	3	4	5	6	7
o						

player 2 drop where?

1	2	3	4	5	6	7
o				@		

player 1 drop where?

1 - - - - -

1	2	3	4	5	6	7
o						
o				@		

player 2 drop where?

1	2	3	4	5	6	7
o						
o						
o	@			@		

player 2 drop where?

1	2	3	4	5	6	7
o						
o						
o	@			@		

player 2 drop where?

1	2	3	4	5	6	7
---	---	---	---	---	---	---

	1	2	3	4	5	6	7
	o						
	o						
	o	@	@		@		

player 1 drop where?

2	1	2	3	4	5	6	7
---	---	---	---	---	---	---	---

o						
o	o					
o	@	@		@		

player 2 drop where?

4 1 2 3 4 5 6 7

o						
o	o					
o	@	@	@	@		

```

Player 2 WON!
Press ENTER to continue...

```

# TEST CASES Vertical Win

1	2	3	4	5	6	7

player 1 drop where?

1	2	3	4	5	6	7
O						

player 2 drop where?

1	2	3	4	5	6	7
O	@					

player 1 drop where?

1	2	3	4	5	6	7
---	---	---	---	---	---	---

1	2	3	4	5	6	7
o						
o	@					

player 2 drop where?

2

1	2	3	4	5	6	7
o	@					
o	@					

player 1 drop where?

1

1	2	3	4	5	6	7
o						
o	@					
o	@					

player 2 drop where?

2



1	2	3	4	5	6	7
o						
o	@					
o	@					
o	@					

Player 1 WON !  
Press ENTER to continue...

## Test Case With Non-Valid Input

1

1	2	3	4	5	6	7
@						
o						
@						
o						
@						
o						

player 1 drop where?  
1  
The Column Is Full, Please Enter Another Number :D  
13  
not Valid Number!  
please enter a value between 1 and 7 :\_