Prabhu N.D. – 99946 75750 | 962 962 0432
www.sixphrase.com | sixphrase@gmail.com
www.facebook.com/SixPhrase

1) **Problem:** There is a colony of 8 cells arranged in a straight line where each day every cell competes with its adjacent cells(neighbour). Each day, for each cell, if its neighbours are both active or both inactive, the cell becomes inactive the next day, otherwise it becomes active the next day.

Assumptions: The two cells on the ends have single adjacent cell, so the other adjacent cell can be assumed to be always inactive.
Even after updating the cell state. consider its previous state for updating the state of other cells. Update the cell information of all cells simultaneously.
Write a function cellCompete which takes takes one 8 element array of integers cells representing the current state of 8 cells and one integer days representing te number of days to simulate.
An integer value of 1 represents an active cell and value of 0 represents an inactive cell.

program:

```
int* cellCompete(int* cells,int days)
{/
/write your code here
}
//function signature ends
```

TESTCASES 1:
INPUT:
[1,0,0,0,0,1,0,0],1
EXPECTED RETURN VALUE:
[0,1,0,0,1,0,1,0]
TESTCASE 2:
INPUT:

[1,1,1,0,1,1,1,1,],2

EXPECTED RETURN VALUE:

[0,0,0,0,0,1,1,0]

2) **Question 2:** Write a function to insert an integer into a circular linked _list whose elements are sorted in ascending order 9smallest to largest). The input to the function insertSortedList is a pointer start to some node in the circular list and an integer n between 0 and 100. Return a pointer to the newly inserted node.

The structure to follow for a node of the circular linked list is_

Struct CNode ;

Typedef struct CNode cnode;

Struct CNode

{I

nt value;

Cnode next;*

};C

node insertSortedList (cnode* start,int n*

{/

/WRITE YOUR CODE HERE

}/

/FUNCTION SIGNATURE ENDS

TestCase 1:

Input:

[3>4>6>1>2>^],5

Expected Return Value:

[5>6>1>2>3>4>^]

TestCase 2:

Input:

[1>2>3>4>5>^],0

Expected Return Value:

[0>1>2>3>4>5>^]

3) **Question 3:** The LeastRecentlyUsed(LRU) cache algorithm exists the element from the cache(when it's full) that was leastrecentlyused. After an element is requested from the cache, it should be added to the cache(if not already there) and considered the most recently used element in the cache. Initially, the cache is empty. The input to the function LruCountMiss shall consist of an integer max_cache_size, an array pages and its length len.
The function should return an integer for the number of cache misses using the LRU cache algorithm.
Assume that the array pages always has pages numbered from 1 to 50.

TEST CASES:

TEST CASE1:

INPUT:

3,[7,0,1,2,0,3,0,4,2,3,0,3,2,1,2,0],16

EXPECTED RETURN VALUE:

11

TESTCASE 2:

INPUT:

2,[2,3,1,3,2,1,4,3,2],9

EXPECTED RETURN VALUE:

8

EXPLANATION:

The following page numbers are missed one after the other 2,3,1,2,1,4,3,2.This results in 8 page misses.

CODE:

```
int lruCountMiss(int max_cache_size, int *pages,int len)
{/
/write tour code
}
```

4) **Question 4:** A system that can run multiple concurrent jobs on a single CPU have a process of choosing which task hast to run when, and how to break them up, called "scheduling". The RoundRobin policy for scheduling runs each job for a fixed amount of time before switching to the next job. The waiting time fora job is the total time that it spends waiting to be run. Each job arrives at particular time for scheduling and certain time to run, when a new job arrives, It is
scheduled after existing jobs already waiting for CPU time Given list of job submission, calculate the average waiting time for all jobs using RoundRobin policy.
The input to the function **waitingTimeRobin** consist of two integer arrays containing job arrival and run times, an integer n representing number of jobs and am integer q representing the fixed amount of time used by RoundRobin policy. The list of job arrival time and run time sorted in ascending order by arrival time. For jobs arriving at same time, process them in the order they are found in the arrival array. You can assume that jobs arrive in such a way that CPU is never idle.
The function should return floating point value for the average waiting time which is calculated using round robin policy.
Assume 0<=jobs arrival time < 100 and 0<job run time <100.

5) **Problem:**
Mooshak the mouse has been placed in a maze.There is a huge chunk of cheese somewhere in the maze. The maze is represented as a two dimensional array of integers, where 0 represents walls.1 repersents paths where mooshak can move and 9 represents the huge chunk of cheese.

Mooshak starts in the top left corner at 0.
Write a method is Path of class Maze Path to determine if Mooshak can reach the huge chunk of cheese. The input to is Path consists of a two dimensional array gnd for the maze matrix. the method should return 1 if there is a path from Mooshak to the cheese.and 0 if not Mooshak is not allowed to leave the maze or climb on walls.

EX: 8 by 8(8*8) matrix maze where Mooshak can get the cheese.

```
1 0 1 1 1 0 0 1

1 0 0 0 1 1 1 1

1 0 0 0 0 0 0 0

1 0 1 0 9 0 1 1

1 1 1 0 1 0 0 1

1 0 1 0 1 1 0 1

1 0 0 0 0 1 0 1

1 1 1 1 1 1 1 1
```

Test Cases:

Case 1:

Input:[[1,1,1,][9,1,1],[0,1,0]]

Expected return value :1

Explanation:

The piece of cheese is placed at(1,0) on the grid Mooshak can move from (0,0) to (1,0) to reach it or can move from (0,0) to (0,1) to (1,1) to (1,0)

Test case 2:

Input: [[0,0,0],[9,1,1],[0,1,1]]

Expected return value: 0

Explanation:

Mooshak cannot move anywhere as there exists a wall right on (0,0)

6) Test whether an input string of opening and closing parentheses was balanced or not. If yes , return the no. of parentheses. If no, return -1.

7) Reverse the second half of an input linked list. If the input linked list contained odd number of elements , consider the middlemost element too in the second half.

8) Write a program to Merge sort using pointers

9) Merge two sorted singly linked lists into one sorted list

10) Reverse a linked list using recursion and without recursion

11) Removal of vowel from string

12) Print and count all the numbers which are less than a given key element from a given array.

13) Eliminate repeated letters in Array

14) String Reversal

15) Find a target value in a two-dimensional matrix given the number of rows as rowCount and number of columns as columnCount, and return its coordinates. If the value didn't exist, the program had to return (-1,-1).

16) Obtain a output as follows:

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
11 12 13 14 15
7 8 9 10
4 5 6
2 3
1

17) Write a C program to print below,when n=4 and s=3

3
44
555
555
44
3

18) Input: num1 and num2 such that 0 <= num1 <= 99999999 and 0 <= num2 <= 9. You have to find number of occurences of input num2 in input num1 and return it with function int isOccured(int num1, int num2).

Example:

Input: num1= 199294, num2= 0

Output: 3

Test Case 1:

Input:

1222212

2

Expected Output:

5

Test Case 2:

Input:

1001010

0

Expected Output:

4

19) Find a sub string in a given string and replace it with another string?
20) Remove all the vowels from a given string using pointers concept
21) Program to check for prime number:
22) To find GCD of two number.
23) Find the GCD of N numbers using pointers
24) Print all the prime numbers which are below the given number separated by comma.
25) C program to find out prime factors of given number
26) C Program to Display Prime Numbers between Two Intervals
27) C Program to Display Armstrong Number Between Two Intervals
28) Write a function to return a sorted array after merging two unsorted arrays, the parameters will be two integer pointers for referencing arrays and two int variable, the length of arrays (Hint: use malloc() to allocate memory for 3rd array):
29) Get an unsorted array and convert into alternate array(alternate array-ascending order array by taking alternate elements).Half elements of array in ascending remaining half in descending.The first n elements should be sorted in ascending order and the next part should be sorted in descending and print it

Test Case Input: [1,5,6,8,4,7]6; 6 is the length of array

30) Given 5,1,4,7,9....do alternate sort for this..and print 1,5,9

31) Pattern Question:- For n=5

1
3*2
4*5*6
10*9*8*7
11*12*13*14*15

32) Pattern Question:-

1
22
333
4444
55555
4444
333
22
1

33) Pattern Question:-

To print the pattern like for n=3 the program should print

1 1 1 2
3 2 2 2
3 3 3 4

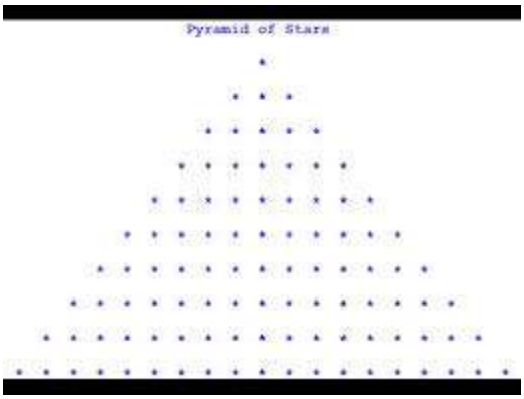
34) print n=6; n stands for number of lines

1111112
3222222
3333334
5444444
5555556
7666666

35) To print the trapezium pattern. for example , we have n=4 the output should be like

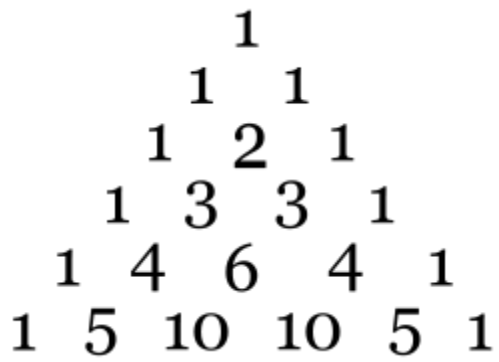
1*2*3*4*17*18*19*20
- -5*6*7*14*15*16
- - - -8*9*12*13
- - - - - -10*11

36) C program to print following pyramid pattern of stars



37) Write a c program to print Pascal triangle.

In mathematics, Pascal's triangle is a triangular array of the binomial coefficients. In much of the Western world it is named after French mathematician Blaise Pascal, although other mathematicians studied it centuries before him in India, Iran, China, Germany, and Italy.



38) 1
3*2
4*5*6
9*8*7
14*13*12*11

39) 1
2*3
4*5*6
7*8*9*10
7*8*9*10
4*5*6
2*3
1

40) 1

2 2
3 3 3
4 4 4 4

41) 1*2*3*4
9*10*11*12
13*14*15*16
5*6*7*8