

|  |
| --- |
|  |
| Onsite Programming Contest |

|  |  |  |  |
| --- | --- | --- | --- |
| QR code |  |  |  |
| K! ID |  |  |  |
| MOBILE NUMBER |  |  |  |
| NAME |  |  |  |
| COLLEGE |  |  |  |

**RULES**

* A team can have a minimum of one member and maximum of three members.
* The participants must have a valid K! ID to participate .
* The team members need not necessarily be from same institution/college.
* A participant cannot be a member of more than one team.
* The participants are not allowed to bring any additional material.
* Teams involved in any kind of malpractice will be disqualified immediately.Decisions made by the administrators will be final.

**Part A**

1)

#include <stdio.h>

int main()

{

static int var[5];

int count=0;

var[++count]=++count;

for(count=0;count<5;count++)

printf("%d ",var[count]);

return 0;

}

--------------------------------------------------

2)

main()

{

char s[ ]="man";

int i;

for(i=0;s[ i ];i++)

printf("\n%c%c%c%c",s[ i ],\*(s+i),\*(i+s),i[s]);

}

--------------------------------------------------

3)

main()

{

char \*p;

printf("%d %d ",sizeof(\*p),sizeof(p));

}

--------------------------------------------------

4)

main()

{

int i=5;

printf("%d%d%d%d%d%d",i++,i--,++i,--i,i);

}

--------------------------------------------------

5)

main( )

{

static int a[ ] = {0,1,2,3,4};

int \*p[ ] = {a,a+1,a+2,a+3,a+4};

int \*\*ptr = p;

ptr++;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

\*ptr++;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

\*++ptr;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

++\*ptr;

printf(“\n %d %d %d”, ptr-p, \*ptr-a, \*\*ptr);

}

--------------------------------------------------

6)

#define FALSE -1

#define TRUE 1

#define NULL 0

main() {

if(NULL)

puts("NULL");

else if(FALSE)

puts("TRUE");

else

puts("FALSE");

}

--------------------------------------------------

7)

main()

{

signed char i=0;

for(;i>=0;i++) ;

printf("%d\n",i);

}

--------------------------------------------------

8)

main()

{

int i=10,j=20;

j = i, j?(i,j)?i:j:j;

printf("%d %d",i,j);

}

--------------------------------------------------

9)

#include <stdio.h>

void fun(int \*ptr)

{

\*ptr=100;

}

int main()

{

int num=50;

int \*pp=#

fun(& \*pp);

printf("%d,%d",num,\*pp);

return 0;

}

--------------------------------------------------

10)

#include <stdio.h>

int main()

{

int a=10;

if(a==10)

{

printf("Hello...");

break;

printf("Ok");

}

else

{

printf("Hii");

}

return 0;

}

--------------------------------------------------

11)

#include <stdio.h>

int main()

{

int c = 4;

c = c++ + ~++c;

printf("%d", c);

return (0);

}

12)

#include <stdio.h>

int main()

{

int n, i = 5;

n = - - i--;

printf("%d%d", n, i);

return (0);

}

--------------------------------------------------

13)

main()

{

int a, b, c;

a = 10;

b = 20;

c = printf("%d",a) + ++b;

printf ("%d",c);

}

--------------------------------------------------

14)

#include<stdio.h>

int main()

{

int a = 320;

char \*ptr;

ptr = (char \*)&a;

printf("%d",\*ptr);

return 0;

}

--------------------------------------------------

15)

main()

{

if(1,0){

printf("True");

}

else{

printf("False");

}

}

--------------------------------------------------

**Part B**

1) Given an array of length n, check whether it is beautiful array or not.

An array is considered as beautiful when for any two numbers a[i] and a[j],

a[i]\*a[j]=a[k],where k=i or k=j.

2) In a doubly linked list, value of the node is in accending order upto a certain node and after it goes in decending order. Make the doubly linked list as sorted list.

Sample:

Input: 2<->5<->11<->23<->14<->8<->3

Output: 2<->3<->5<->8<->11<->14<->23

3) Print the binary tree in vertical order.

# 4) Given an expression with only ‘{‘ and ‘}’,find the minimum number of bracket reversals needed to make it balanced.

Input: exp = "{{{"

Output: Can't be made balanced using reversals

Input: exp = "{{{{"

Output: 2

5) Find the next permutation of a given string.

(Note : Try to avoid using STL)

Eg:

Input : hello

Output : helol

6) You are a professional robber planning to rob houses along a street. Each house has a certain amount of money stashed. All houses at this place are **arranged in a circle.** That means the first house is the neighbor of the last one. Meanwhile, adjacent houses have security system connected and it will automatically contact the police if two adjacent houses were broken into on the same night.

Given a list of non-negative integers representing the amount of money of each house, determine the maximum amount of money you can rob tonight **without alerting the police**

Eg:

Input : [2,8,6,3,4]

Output : 12

7) Given a string containing just the characters ‘)’ and ‘(‘, find the length of the longest valid parentheses substring.

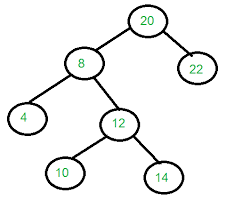
Eg:

Input: ( ( ) ( ) ( ( )

Output: 4

8) Print “Hello world” n^n times, where n ranges from 0 to INT\_MAX.

9) Given a Binary tree structure as pre-order representation in an array .If the input is -1, it considered as NULL.



Example : 20,8,22,4,12,-1,-1,-1,12,10,14

Consider root of tree is 20:

Print in-order traversal of the tree without using tree data Structure.

10) Print the pre-order traversal of a binary tree, without using array and recursion.