**MST -2**

**2nd SEMESTER**

**CS101-Introduction to C Programming**

**Set- B**

**Time allowed: 120 Minutes Max. Marks: 30**

**General Instructions:**

* **All Questions are mandatory**

1. Predict the output of the code

void main()

{

int a[10];

printf("%d %d", a[-1], a[12]);

}

1. 0 0
2. **Junk junk**
3. 0 junk
4. Junk 0
5. Which of the following declaration are illegal?
6. **int a[][] = {{1, 2, 3}, {2, 3, 4, 5}};**
7. int \*a[] = {{1, 2, 3}, {2, 3, 4, 5}};
8. int a[4][4] = {{1, 2, 3}, {2, 3, 4, 5}};
9. int a[][2]={1,2};

|  |
| --- |
| 1. In C, if you pass an array as an argument to a function, what actually gets passed? |
| |  |  | | --- | --- | | [**A.**](javascript:%20void%200;) | Value of elements in array | | [**B.**](javascript:%20void%200;) | First element of the array | | [**C.**](javascript:%20void%200;) | **Base address of the array** | | [**D.**](javascript:%20void%200;) | Address of the last element of array | |

4. What is the output of this C code?

#include<stdio.h>

int main()

{

static int x;

printf("%d", x);

}

**a) 0**

b) 1

c) Garbage Value

d) None of the above

1. Choose the correct option for statement 1 and 2

1. int (\*fun)(int);

2. int \*fun (int);

a) 1. Function to pointer

2. pointer to function

**b) 1. Function pointer**

**2. function with return type int\***

c) both 1 and 2 are function pointers

d) both 1 and 2 are functions with return type int\*

1. What is the output of the following code?

int a[] = {1,2,3,4,5};

for(int i=0; i<5; i++)

printf("%d ",a[i]++);

1. **1 2 3 4 5**
2. 2 3 4 5 6
3. 2 3 4 5 5
4. 1 3 4 5 6

**7.** What will the output of following code.

#include <stdio.h>

void fun()

{

static int i=5;

if(--i)

printf("%d ",i);

}

int main()

{

int i;

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

fun();

}

1. **4 3 2 1**
2. 5 5 5 5
3. 4 3 2 1 0
4. 5 5 5 5 5

8. What will be the output of the following code

#include <stdio.h>

#define MAX\_SIZE 100

void edit\_string(char \* str);

int main()

{

char str[MAX\_SIZE]="Chitkara University";

edit\_string(str);

printf("%s", str);

return 0;

}

void edit\_string(char \* str)

{

while(\*str)

{ if(\*str >= 'a' && \*str <= 'z')

\*str = \*str - 32;

else if(\*str >= 'A' && \*str <= 'Z')

\*str = \*str + 32;

str++;

}

}

1. CHITKARA UNIVERSITY
2. Chitkara university
3. **cHITKARA uNIVERSITY**
4. Chitkara University

9. Identify the correct output for the following code:

int a[5] = {1,2,3,4,5};

int \*ptr = a;

printf(“%d %d ”, \*a, \*ptr);

ptr++;

printf(“%d %d”, \*a, \*ptr);

1. 5 5 5 6
2. 1 1 2 2
3. **1 1 1 2**
4. 5 1 5 2

10. Choose the correct option

#include<stdio.h>

static int k = 5;

int main()

{

int sum = 0;

do

{

sum+=(1/k);

}while(0<k--);

printf("sum of the series is %d",sum);

return 0;

}

1. It will get executed fine
2. It will produce a compile time error
3. **It will produce a run time error**
4. None of these

**Coding-1 (5 marks)**

The kindergarten School planned evaluation for their Nursery students which will be combination of their English and Math skills.AS at this stage students have knowledge of numbers from 0 to 50 and about 26 alphabets of English(uppercase and lowercase).The task was to evaluate students about alphabet recognition and counting.so given a string ,and a character,the task is to check whether given character is present in string or not.if yes ,then tell the index at which character is present in string by starting counting from 0.

**Sample Input 1**

**Ramesh**

**A**

**Sample output 1**

**'A' not found.**

**Sample Input 2**

**vinita**

**i**

**Sample output 2**

**'i' is found at index 1.**

**Test Cases:**

**Input 1**

**Aggarwal**

**a**

**output 1**

**'a' is found at index 3.**

**Input 2**

**M0na**

**0**

**Output 2**

**'0' is found at index 1.**

**Input 3**

**CS101**

**O**

**Output 3**

**'O' not found.**

**Code stub:**

**#include<stdio.h>**

**#include <string.h>**

**#define MAX\_SIZE 50**

**int indexOf(const char \* str, const char toFind);**

**int main()**

**{**

**char str[MAX\_SIZE];**

**char toFind,to;**

**int index;**

**/\* Input string from user and character to be searched \*/**

**//printf("Enter any string: ");**

**scanf("%s",str);**

**to=getchar();**

**//printf("Enter character to be searched: ");**

**toFind = getchar();**

**index = indexOf(str, toFind);**

**if(index == -1)**

**printf("'%c' not found.", toFind);**

**else**

**printf("'%c' is found at index %d.", toFind, index);**

**return 0;**

**}**

**Solution:**

int indexOf(const char \* str, const char toFind)

{

int i = 0;

while(str[i] != '\0')

{

if(str[i] == toFind)

return i;

i++;

}

// Return -1 as character not found

return -1;

}

**Coding-2(10 marks)**

Anny has been appointed on her new job. She has been assigned a very confusing task with bulk of data. There’s a high probability of error and the chances of missing the deadline are very high. Help Anny to sort out the data with the help of C code:

Develop a code which accepts the marks of students and stores it in an array. Sort the marks in descending order. Extract the maximum and minimum marks from the given list. Find the difference between maximum and minimum marks and subtract this difference from the marks of every student and then display the new list of marks.

Note: Elements in array must be between 1-10. If value of n comes lesser than 1 or greater than 10 then program terminates with error message “**enter correct value of n**”.

**Input:**

Consists of n+1 integers.

n //number of elements in an array

enter n elements of array.

**Output:**

Array elements after performing desired operation

**Sample Input 1:**

4 //number of elements in array

2

1

4

3

**Sample Output 1:**

1

0

-1

-2

**Explanation:** Array elements are: 2 1 4 3 .So firstly sort in descending order it becomes 4 3 2 1,now find max and min element. Max is 4 and min is 1. Difference of max and min is 4-1=3.Now subtract difference form all elements ie.

4-3=1

3-3=0

2-3=-1

1-3=-2

So output is 1 0 -1 -2

**Sample Input 2:**

11 //number of elements in array

**Sample Output 2:**

enter correct value of n

**Test cases:**

**Test Case 1: (2 MARKS)**

INPUT:

4

126

110

200

141

**OUTPUT:**

110

51

36

20

**Test Case 2: (2 MARKS)**

**INPUT:**

5 //no. of elements

-2

-6

-4

-11

-81

**OUTPUT:**

-81

-83

-85

-90

-160

**Test Case 3: (2 mark)**

**INPUT:**

0

**OUTPUT:**

enter correct value of n

**Test Case 4: (2 mark)**

**INPUT:**

15

**OUTPUT:**

enter correct value of n

**Test Case 5: (2 mark)**

**INPUT:**

2 //no. of elements

3

11

**OUTPUT:**

3

-5

**Code Stub:**

**#include<stdio.h>**

**void read(int arr[],int);**

**void revsort(int arr[],int);**

**int diff(int arr[],int);**

**int min(int arr[],int);**

**int max(int arr[],int);**

**void display(int arr[],int,int);**

**int main()**

**{**

**int arr[30],n,diff1;**

**scanf("%d",&n);**

**if((n>=1) && (n<=10))**

**{**

**read(arr,n);**

**revsort(arr,n);**

**diff1=diff(arr,n);**

**display(arr,n,diff1);**

**}**

**Solution:**

else

{

printf("enter correct value of n");

}

return 0;

}

void read(int \*a,int n)

{

int i;

for(i=0;i<n;i++)

{

scanf("%d",&a[i]);

}

}

void revsort(int \*a,int n)

{

int i,j,temp;

for(i=0;i<n-1;i++)

{

for(j=0;j<n-i-1;j++)

{

if(a[j]<a[j+1])

{

temp=a[j];

a[j]=a[j+1];

a[j+1]=temp;

} } }}

int diff(int \*a,int n)

{

int min1,max1,diff1;

min1=min(a,n);

max1=max(a,n);

diff1=max1-min1;

return(diff1);

}

int min(int \*a,int n)

{

int min1,i;

min1=a[0];

for(i=0;i<n;i++)

{

if(a[i]<min1)

{

min1=a[i];

}

}

return min1;

}

int max(int \*a,int n)

{

int max1,i;

max1=a[0];

for(i=0;i<n;i++)

{

if(a[i]>max1)

{

max1=a[i];

} }

return max1;

}

void display(int \*a,int n,int diff)

{

int i;

for(i=0;i<n;i++)

{

printf("\n%d",a[i]-diff);

}}