**FA3**

**MCQs of 1 Mark each**

Que 1. What is the output of the following program?

#include<stdio.h>

int tmp=20;

int main( )

{

printf("%d ",tmp);

func( );

printf("%d ",tmp);

}

void func( )

{

static int tmp=10;

printf("%d ",tmp);

}

a) 20 10 10

**b) 20 10 20**

c) 20 20 20

d) 10 10 10

Ques 2 Consider one-dimensional array: int values[]={11,22,33,44,55,66,77,88,99} which has base address = 2020 and the word size = 4. Now compute the address of the element values[3].

a)  2026

b) 2036

**c) 2032**

d) 2040

Que 3 What will be the output of following code :

int main()

{

int arr1[]={11,11};

int arr2[]={1,2,3,4};

int arr3[]={9,8,7,6,5};

int \*parr[3]={arr1,arr2,arr3};

int i;

for (i=0;i<3;i++)

{

printf("%d ",\*parr[i]);

}

return 0;

}

a) 11 11 1 2 3 4 9 8 7 6 5

b) 11 11 1

**c) 11 1 9**

d) none of these

Que 4. Consider a two-dimensional array (arr) which is stored in memory in row major order. The expression arr+1 will point to:

a) arr[0][1]

**b) arr[1][0]**

c) arr[1][1]

d) arr[0][0]

**MCQs of 2 Marks each**

|  |
| --- |
| Que 5  #include <stdio.h>  int main()  {    int x = 10;    static int y = x;      if(x = = y)       printf("Equal");    elseif(x > y)       printf("Greater");    else       printf("Less");    return0;  } |

**(A) Compiler error**

(B) Equal  
(C) Greater  
(D) Less

Que 6

#include <stdio.h>

int main()

{

    float arr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};

    float \*ptr1 = &arr[0];

    float \*ptr2 = ptr1 + 3;

    printf("%f ", \*ptr2);

    printf("%d", ptr2 - ptr1);

   return 0;

}

**A 90.500000 3**

B 90.500000 12

C 10.000000 12

D 0.500000 3

Que 7

#include <stdio.h>

void f(int a[][3])

{

a[0][1]=3;

int i=0, j =0;

for(i=0;i<2;i++)

for(j =0; j <3;j++)

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

}

void main()

{

int a[2][3]={0};

f(a);

}

**a) 0 3 0 0 0 0**b) Junk 3 junk junkjunkjunk  
c) Compile time error  
d) All junk values

**Coding Question of 10 mark**

Que 1. Create a array of n elements. Ask elements from the user. Then find prime numbers in array and store all prime numbers in 2nd array.

Print all these prime numbers and sum of these prime numbers

Sample input1:-

7 // n number of elements

4 //array

3

1

2

5

8

9

Sample output1 :-

Prime Numbers are-

3

2

5

Sum= 10

Sample input 2:

5

3

3

4

5

6

Sample output 2:

Prime Numbers are-

3

3

5

Sum=11

**5 Testcases- 2marks each**

**Input Test case 1:**

10

77

41

34

11

9

71

67

45

33

53

**Output test case1:**

Prime Numbers are-

41

11

71

67

53

Sum=243

**Input test case 2:**

4

1

4

5

8

**Output Test case 2:**

Prime Numbers are-

5

Sum=5

**Input test case 3:**

5

1

3

5

2

4

**Output test case 3:**

Prime Numbers are-

3

5

2

Sum=10

**Input test case 4:**

12

41

31

22

1

5

61

71

98

55

64

5

7

**Output test case 4:**

Prime Numbers are-

41

31

5

61

71

5

7

Sum=221

**Input test case 5:**

7

223

501

31

11

90

409

321

**Output test case 5:**

Prime Numbers are-

223

31

11

409

Sum=674

Solution:-

#include<stdio.h>

int main()

{

int a[100],n,i,j,c=0,prime[100],t=0,s=0;

scanf("%d",&n);

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

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

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

{

c=0;

for(j=2;j<a[i];j++)

{

if(a[i]==1)

{

c=1;

break;

}

if(a[i]%j==0)

{

c=1;

break;

}

}

if(c==0&&a[i]!=1)

{

prime[t]=a[i];

t++;

}

}

printf("Prime Numbers are-\n");

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

{

printf(" %d\n ",prime[i]);

s+=prime[i];

}

printf("Sum=%d",s);

return 0;

}