LAB REPORT (SE122)

ID:221-35-858

1. Write a program in C to store elements in an array and print it

2. Write a program in C to read n number of values in an array and display it in reverse order.

3. Write a program in C to find the sum of all elements of the array

4. Write a program in C to copy the elements of one array into another array

5. Write a program in C to count a total number of duplicate elements in an array.

6. Write a program in C to print all unique elements in an array

7. Write a program in C to merge two arrays of same size sorted in descending order

8. Write a program in C to find the maximum and minimum element in an array.

9. Write a program in C to separate odd and even integers in separate arrays

10. Write a program in C to sort elements of array in ascending order.

11. Write a program in C to sort elements of the array in descending order.

12. Write a program in C to insert New value in the array (unsorted list).

13. Write a program in C to delete an element at desired position from an array.

14. Write a program in C to find the second largest element in an array.

15. Write a program in C to find the second smallest element in an array

16. Write a program in C for a 2D array of size 3x3 and print the matrix.

17. Write a program in C for addition of two Matrices of same size.

18. Write a program in C to display the n terms of odd natural number and their sum.

19. Write a program in C to display the n terms of harmonic series and their sum.

20. Write a C program to determine whether a given number is prime or not.

21. Write a program in C to find the number and sum of all integer between 100 and 200 which are divisible by 9.

22. Write a program in C to find the sum of the series 1 +11 + 111 + 1111 + ….. n terms

23.Write the code to find the factorial of that number.

24.Enter a six digit number and print the number in reverse order and find the sum of those number.

25.Write a menu driven program which has the following options.

i)Factorial

ii) Prime or not

iii) odd

iv) Even

v) Exit

Solve-1:

```
#include <stdio.h>
int main()
{
	int n;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++)
	{
		scanf("%d", &a[i]);
		printf("%d ", a[i]);
	}
	return 0;
}
```

Solve-2:

```
#include <stdio.h>
int main()
{
	int n;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	printf("Displaying array in reverse order: ");
	for (int i = n - 1; i >= 0; i--) printf("%d ", a[i]);
	return 0;
}
```

Solve-3:

```
#include <stdio.h>
int main()
{
	int sum = 0, a[5] = { 45, 76, 99, 24, 86 };
	printf("Array elements: ");
	for (int i = 0; i < 5; i++)
	{
		printf("%d ", a[i]);
		sum += a[i];
	}
	printf("\nSum of it's elements: %d\n", sum);
	return 0;
}
```

Solve-4:

```
#include <stdio.h>
int main()
{
	int A[5] = { 45, 76, 99, 24, 86}, B[5];
	printf("After copying elements from array A to B.\n");
	for (int i = 0; i < 5; i++)
	{
		B[i] = A[i];
		printf("%d ", B[i]);
	}
	return 0;
}
```

Solve-5:

```
#include <stdio.h>
int main()
{
	int n, c = 0;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	for (int i = 0; i < n - 1; i++)
	{
		for (int j = i + 1; j < n; j++)
		{
			if (a[i] == a[j])
			{
				c++;
				break;
			}
		}
	}
	printf("No. of duplicate element %d\n", c);
	return 0;
}
```

Solve-6:

```
#include <stdio.h>
int main()
{
	int n;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	printf("Unique elements: ");
	for (int i = 0; i < n; i++)
	{
		int c = 1;
		for (int j = 0; j < n; j++)
		{
			if (a[i] == a[j] && i != j)
			{
				c = 0;
				break;
			}
		}
		if (c) printf("%d ", a[i]);
	}
	return 0;
}
```

Solve-7:

```
#include <stdio.h>
int main()
{
	int temp, x[10], a[5] = { 7, 3, 6, 9, 3}, b[5] = { 11, 0, 2, 5, 8 };
	for (int i = 0; i < 5; i++)
	{
		x[i] = a[i];
		x[i + 5] = b[i];
	}
	for (int i = 0; i < 10; i++)
	{
		for (int j = i + 1; j < 10; j++)
		{
			if (x[i] < x[j])
			{
				temp = x[i];
				x[i] = x[j];
				x[j] = temp;
			}
		}
	}
	printf("After merging two arrays, displaying it in descending order:
");
	for (int i = 0; i < 10; i++) printf("%d ", x[i]);
	return 0;
}
```

Solve-8:

```
#include <stdio.h>
int main()
{
	int a[5] = { 14, 5, 11, 9, 18 }, max = a[0], min = a[0];
	for (int i = 0; i < 5; i++) printf("%d ", a[i]);
	for (int i = 1; i < 5; i++)
	{
		if (max < a[i]) max = a[i];
		if (min > a[i]) min = a[i];
	}
	printf("\nMaximum element: %d\n", max);
	printf("Minimum element %d\n", min);
	return 0;
}
```

Solve-9:

```
#include <stdio.h>
int main()
{
	int x, n = 0, m = 0, odd[100], even[100];
	while (scanf("%d", &x) && x != 0)
	{
		if (x % 2 == 0)
		{
			even[n] = x;
			n++;
		}
		else
		{
			odd[m] = x;
			m++;
		}
		printf("%d\n", x);
	}
	printf("Array of even elements: ");
	for (int i = 0; i < n; i++) printf("%d ", even[i]);
	printf("\nArray of odd elements: ");
	for (int i = 0; i < m; i++) printf("%d ", odd[i]);
	return 0;
}
```

Solve-10:

```
#include <stdio.h>
int main()
{
	int n, temp;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	for (int i = 0; i < n - 1; i++)
	{
		for (int j = i + 1; j < n; j++)
		{
			if (a[i] > a[j])
			{
				temp = a[i];
				a[i] = a[j];
				a[j] = temp;
			}
		}
	}
	printf("Sorted array in ascending order: ");
	for (int i = 0; i < 5; i++) printf("%d ", a[i]);
	return 0;
}
```

Solve-11:

```
#include <stdio.h>
int main()
{
	int n, temp;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	for (int i = 0; i < n - 1; i++)
	{
		for (int j = i + 1; j < n; j++)
		{
			if (a[i] < a[j])
			{
				temp = a[i];
				a[i] = a[j];
				a[j] = temp;
			}
		}
	}
	printf("Sorted array in descending order: ");
	for (int i = 0; i < 5; i++) printf("%d ", a[i]);
	return 0;
}
```

Solve-12:

```
#include <stdio.h>
int main()
{
	int n, element, pos;
	printf("Enter array size: ");
	scanf("%d", &n);
	int arr[n + 1];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &arr[i]);
	printf("Enter the position and the new element: ");
	scanf("%d%d", &pos, &element);
	for (int i = n; i > pos - 1; i--) arr[i] = arr[i - 1];
	arr[pos - 1] = element;
	printf("Resultant array after insertion: ");
	for (int i = 0; i <= n; i++) printf("%d ", arr[i]);
	return 0;
}
```

Solve-13:

```
#include <stdio.h>
int main()
{
	int n, pos;
	printf("Enter array size: ");
	scanf("%d", &n);
	int arr[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &arr[i]);
	printf("Enter the position where you wish to delete: ");
	scanf("%d", &pos);
	for (int i = pos - 1; i < n; i++) arr[i] = arr[i + 1];
	printf("Resultant array after deletion: ");
	for (int i = 0; i < n - 1; i++) printf("%d ", arr[i]);
	return 0;
}
```

Solve-14:

```
#include <stdio.h>
int main()
{
	int n, temp;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	for (int i = 0; i < n - 1; i++)
	{
		for (int j = i + 1; j < n; j++)
		{
			if (a[i] < a[j])
			{
				temp = a[i];
				a[i] = a[j];
				a[j] = temp;
			}
		}
	}
	for (int i = 1; i < 5; i++)
	{
		if (a[0] > a[i])
		{
			printf("Second largest element: %d\n", a[i]);
			break;
		}
	}
	return 0;
}
```

Solve-15:

```
#include <stdio.h>
int main()
{
	int n, temp;
	printf("Enter array size: ");
	scanf("%d", &n);
	int a[n];
	printf("Enter array elements: ");
	for (int i = 0; i < n; i++) scanf("%d", &a[i]);
	for (int i = 0; i < n - 1; i++)
	{
		for (int j = i + 1; j < n; j++)
		{
			if (a[i] > a[j])
			{
				temp = a[i];
				a[i] = a[j];
				a[j] = temp;
			}
		}
	}
	for (int i = 1; i < 5; i++)
	{
		if (a[0] < a[i])
		{
			printf("Second smallest element: %d\n", a[i]);
			break;
		}
	}
	return 0;
}
```

Solve-16:

```
#include <stdio.h>
int main()
{
	int a[3][3] = {{ 1, 2, 3 },{ 4, 5, 6 },{ 7, 8, 9 }};
	for (int i = 0; i < 3; i++)
	{
		for (int j = 0; j < 3; j++) printf("%d ", a[i][j]);
		printf("\n");
	}
	return 0;
}
```

Solve-17:

```
#include <stdio.h>
int main()
{
	int a[3][3] = {{ 1, 2, 3 },{ 4, 5, 6 },{ 7, 8, 9 }};
	int b[3][3] = {{ 9, 8, 7 },{ 6, 5, 4 },{ 3, 2, 1 }};
	for (int i = 0; i < 3; i++)
	{
		for (int j = 0; j < 3; j++) printf("%d ", a[i][j] + b[i][j]);
		printf("\n");
	}
	return 0;
}
```

Solve-18:

```
#include <stdio.h>
int main()
{
	int n, sum = 0;
	printf("Enter the range: ");
	scanf("%d", &n);
	printf("Displaying odd numbers in range(%d): ", n);
	for (int i = 1; i <= n; i += 2)
	{
		sum += i;
		printf("%d ", i);
	}
	printf("\nSum of odd numbers in range(%d): %d\n", n, sum);
	return 0;
}
```

Solve-19:

```
#include <stdio.h>
int main()
{
	int n;
	float sum = 0;
	printf("Enter the range: ");
	scanf("%d", &n);
	printf("Displaying harmonic series in range(%d): ", n);
	for (int i = 1; i <= n; i++)
	{
		sum += 1.0 / i;
		if (i == 1) printf("1 ");
		else printf("1/%d ", i);
	}
	printf("\nSum of harmonic series in range(%d): %.1f\n", n, sum);
	return 0;
}
```

Solve-20:

```
#include <stdio.h>
#include <math.h>
int main()
{
	int n, flag = 1;
	printf("Enter a positive integer: ");
	scanf("%d", &n);
	if (n % 2 == 0 && n != 2 || n == 1) flag = 0;
	else
	{
		for (int i = 3; i <= sqrt(n); i += 2)
		{
			if (n % i == 0)
			{
				flag = 0;
				break;
			}
		}
	}
	if (flag)
		printf("%d is a prime number.", n);
	else
		printf("%d is not a prime number.", n);
	return 0;
}
```

Solve-21:

```
#include <iostream>
int main()
{
	int sum = 0;
	printf("Integer between 100 and 200 which are divisible by 9.\n");
	for (int i = 108; i < 200; i += 9)
	{
		sum += i;
		printf("%d ", i);
	}
	printf("\nSum of them: %d\n", sum);
	return 0;
}
```

Solve-22:

```
#include <iostream>
int main()
{
	int n;
	printf("Enter n term: ");
	scanf("%d", &n);
	printf("Sum of the series 1+11+111+..n terms: ");
	for (int i = 1; i <= n; i++) printf("%d", i);
	return 0;
}
```

Solve-23:

```
#include <iostream>
int main()
{
	int n, fact = 1;
	printf("Enter a number: ");
	scanf("%d", &n);
	for (int i = 2; i <= n; i++) fact *= i;
	printf("Factorial of %d is: %d\n", n, fact);
	return 0;
}
```

Solve-24:

```
#include <iostream>
int main()
{
	int sum = 0;
	char d[6];
	printf("Enter a six digit number: ");
	scanf("%s", &d);
	printf("Displaying number in reverse order: ");
	for (int i = 5; i >= 0; i--)
	{
		sum += d[i] - 48;
		printf("%c", d[i]);
	}
	printf("\nSum of it's digit: %d\n", sum);
	return 0;
}
```

Solve-25:

```
#include <stdio.h>
#include <math.h>
int main()
{
	int opt, n, fact = 1, flag = 1;
	printf("1. Factorial\n");
	printf("2. Prime or not\n");
	printf("3. Odd\n");
	printf("4. Even\n");
	printf("5. Exit\n");
	printf("Enter any option: ");
	scanf("%d", &opt);
	switch (opt)
	{
		case 1:
			printf("Enter a number: ");
			scanf("%d", &n);
			for (int i = 2; i <= n; i++) fact *= i;
			printf("Factorial of %d is: %d\n", n, fact);
			break;

		case 2:
			printf("Enter a number: ");
			scanf("%d", &n);
			if (n % 2 == 0 && n != 2 || n == 1) flag = 0;
			else
			{
				for (int i = 3; i <= sqrt(n); i += 2)
				{
					if (n % i == 0)
					{
						flag = 0;
```

```
                                        break;
                                }
                        }
                }
                if (flag)
                        printf("%d is a prime number.", n);
                else
                        printf("%d is not a prime number.", n);
                break;

        case 3:
                printf("Enter a number: ");
                scanf("%d", &n);
                if (n % 2 == 1) printf("Yes, odd.\n");
                else printf("Not, odd.\n");
                break;

        case 4:
                printf("Enter a number: ");
                scanf("%d", &n);
                if (n % 2 == 0) printf("Yes, even.\n");
                else printf("Not, even.\n");
                break;

        case 5:
                printf("Exiting...\n");
                break;

        default:
                printf("Invalid option.\n");
                break;
    }
    return 0;
}
```