

## EXPERIMENT 8:

### SORTING AN ARRAY

#### CODE:

```
#include <stdio.h>
```

```
int main() {
```

```
    int arr[100], n, i, j, temp;
```

```
    printf("Enter number of elements: ");
```

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

```
    printf("Enter %d elements:\n", n);
```

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

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

```
    }
```

```
    // Bubble Sort - Ascending Order
```

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

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

```
            if(arr[j] > arr[j + 1]) {
```

```
                temp = arr[j];
```

```
                arr[j] = arr[j + 1];
```

```
                arr[j + 1] = temp;
```

```
            }
```

```
        }
```

```
    }
```

```
    printf("Sorted array in ascending order:\n");
```

```

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

    printf("%d ", arr[i]);

}

```

```

return 0;

```

```

}

```

OUTPUT:

The screenshot shows the Dev-C++ IDE with a C++ program for sorting an array in descending order. The program prompts the user to enter the number of elements and the elements themselves. It then sorts the array and prints it in descending order. The output window shows the execution results.

```

#include <stdio.h>

int main() {
    int arr[100], n, i, j, temp;
    printf("Enter number of elements: ");
    scanf("%d", &n);
    printf("Enter %d elements:\n", n);
    for(i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }
    for(i = 0; i < n - 1; i++) {
        for(j = 0; j < n - i - 1; j++) {
            if(arr[j] < arr[j + 1]) {
                temp = arr[j];
                arr[j] = arr[j + 1];
                arr[j + 1] = temp;
            }
        }
    }
    printf("Array in descending order:\n");
    for(i = 0; i < n; i++) {
        printf("%d ", arr[i]);
    }
    return 0;
}

```

Output:

```

Enter number of elements: 5
Enter 5 elements:
12
78
95
47
65
Array in descending order:
95 78 65 47 12
Process exited after 16.86 seconds with return value 0
Press any key to continue . . .

```