Experiment 13

CODE:

```
#include <stdio.h>

void displayArray(int arr[], int n) {
    for (int i = 0; i < n; i++)
        printf("%d ", arr[i]);
    printf("\n");
}

void selectionSort(int arr[], int n) {
    int i, j, minIdx, temp;
    for (i = 0; i < n - 1; i++) {
        minIdx = i;
        for (j = i + 1; j < n; j++) {
            if (arr[j] < arr[minIdx])
                minIdx = j;
        }
        temp = arr[minIdx];
        arr[minIdx] = arr[i];
        arr[i] = temp;

        printf("Pass %d: ", i + 1);
        displayArray(arr, n);
        printf("Explanation: In this pass, the minimum element %d was found and 
swapped with element %d.\n", arr[i], arr[minIdx]);
    }
}

void bubbleSort(int arr[], int n) {
    int i, j, temp;
    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("Pass %d: ", i + 1);
        displayArray(arr, n);
        printf("Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.\n");
    }
}

void insertionSort(int arr[], int n) {
    int i, key, j;
    for (i = 1; i < n; i++) {
        key = arr[i];
        j = i - 1;

        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
        arr[j + 1] = key;

        printf("Pass %d: ", i);
        displayArray(arr, n);
        printf("Explanation: In this pass, the element %d was inserted into the sorted section of the array.\n", key);
    }
}

int main() {
    int arr[100], n, choice, i;

    printf("Enter number of elements: ");
    scanf("%d", &n);

    printf("Enter the elements of the array:\n");
    for (i = 0; i < n; i++) {
        scanf("%d", &arr[i]);
    }

    do {
        printf("\nMenu:\n");
        printf("1. Selection Sort\n");
        printf("2. Bubble Sort\n");
        printf("3. Insertion Sort\n");
        printf("4. Exit\n");
        printf("Enter your choice: ");
        scanf("%d", &choice);

        int tempArr[100];
        for (i = 0; i < n; i++) {
            tempArr[i] = arr[i];
        }

        switch (choice) {
            case 1:
                printf("Selection Sort:\n");
                selectionSort(tempArr, n);
                break;
            case 2:
                printf("Bubble Sort:\n");
                bubbleSort(tempArr, n);
                break;
            case 3:
                printf("Insertion Sort:\n");
                insertionSort(tempArr, n);
                break;
            case 4:
                printf("Exiting...\n");
                break;
            default:
                printf("Invalid choice! Please try again.\n");
        }
    } while (choice != 4);

    return 0;
}
```

Output:

```
Enter number of elements: 10
Enter the elements of the array:
12
58
64
20
02
56
```

```
74
34
99
16

Menu:
1. Selection Sort
2. Bubble Sort
3. Insertion Sort
4. Exit
Enter your choice: 1
Selection Sort:
Pass 1: 2 58 64 20 12 56 74 34 99 16
Explanation: In this pass, the minimum element 2 was found and swapped with element 12.
Pass 2: 2 12 64 20 58 56 74 34 99 16
Explanation: In this pass, the minimum element 12 was found and swapped with element 58.
Pass 3: 2 12 16 20 58 56 74 34 99 64
Explanation: In this pass, the minimum element 16 was found and swapped with element 64.
Pass 4: 2 12 16 20 58 56 74 34 99 64
Explanation: In this pass, the minimum element 20 was found and swapped with element 20.
Pass 5: 2 12 16 20 34 56 74 58 99 64
Explanation: In this pass, the minimum element 34 was found and swapped with element 58.
Pass 6: 2 12 16 20 34 56 74 58 99 64
Explanation: In this pass, the minimum element 56 was found and swapped with element 56.
Pass 7: 2 12 16 20 34 56 58 74 99 64
Explanation: In this pass, the minimum element 58 was found and swapped with element 74.
Pass 8: 2 12 16 20 34 56 58 64 99 74
```

```
Explanation: In this pass, the minimum element 64 was found and swapped with element 74.
Pass 9: 2 12 16 20 34 56 58 64 74 99
Explanation: In this pass, the minimum element 74 was found and swapped with element 99.

Menu:
1. Selection Sort
2. Bubble Sort
3. Insertion Sort
4. Exit
Enter your choice: 2
Bubble Sort:
Pass 1: 12 58 20 2 56 64 34 74 16 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 2: 12 20 2 56 58 34 64 16 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 3: 12 2 20 56 34 58 16 64 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 4: 2 12 20 34 56 16 58 64 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 5: 2 12 20 34 16 56 58 64 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 6: 2 12 20 16 34 56 58 64 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 7: 2 12 16 20 34 56 58 64 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 8: 2 12 16 20 34 56 58 64 74 99
```

```
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.
Pass 9: 2 12 16 20 34 56 58 64 74 99
Explanation: In this pass, the largest unsorted element moved to its correct position at the end of the array.


Menu:
1. Selection Sort
2. Bubble Sort
3. Insertion Sort
4. Exit
Enter your choice: 3
Insertion Sort:
Pass 1: 12 58 64 20 2 56 74 34 99 16
Explanation: In this pass, the element 58 was inserted into the sorted section of the array.
Pass 2: 12 58 64 20 2 56 74 34 99 16
Explanation: In this pass, the element 64 was inserted into the sorted section of the array.
Pass 3: 12 20 58 64 2 56 74 34 99 16
Explanation: In this pass, the element 20 was inserted into the sorted section of the array.
Pass 4: 2 12 20 58 64 56 74 34 99 16
Explanation: In this pass, the element 2 was inserted into the sorted section of the array.
Pass 5: 2 12 20 56 58 64 74 34 99 16
Explanation: In this pass, the element 56 was inserted into the sorted section of the array.
Pass 6: 2 12 20 56 58 64 74 34 99 16
Explanation: In this pass, the element 74 was inserted into the sorted section of the array.
Pass 7: 2 12 20 34 56 58 64 74 99 16
Explanation: In this pass, the element 34 was inserted into the sorted section of the array.
Pass 8: 2 12 20 34 56 58 64 74 99 16
```

```
Explanation: In this pass, the element 99 was inserted into the sorted section of the array.
Pass 9: 2 12 16 20 34 56 58 64 74 99
Explanation: In this pass, the element 16 was inserted into the sorted section of the array.


Menu:
1. Selection Sort
2. Bubble Sort
3. Insertion Sort
4. Exit
Enter your choice: 4
Exiting...


=== Code Execution Successful ===
```