Assignment 1: Sorting Techniques

Name: Aarya Gawade
UEC No.: UEC2023122
Batch: A2

Code:

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

int main(){
    int n, i,j, swap, comp, temp, index, small, ch;
    printf("Enter size of array: ");
    scanf("%d", &n);
    int a[n];
    printf("Enter elements of array: \n");
    for(i = 0; i < n; i++){
        scanf("%d", &a[i]);
    }
    printf("Unsorted array: \n");
    for (i = 0; i < n; i++){
        printf("%d\t", a[i]);

    }
    printf("\n");

    //bubble sort
    for(i = 1; i < n; i++) {
        comp += 1;
        for (j = 0; j < n - i; j++){
            swap += 1;
            if( a[j] > a[j+ 1]){
                temp = a[j];
                a[j] = a[j+ 1];
                a[j+ 1] = temp;
```

```
            }

        }

    }




do{
    printf("Menu:\n");
    printf("1. Bubble sort\n");
    printf("2. Insertion sort\n");
    printf("3. Selection sort\n");
    printf("4. Exit\n");
    printf("Enter corresponding number for sorting type or exit: ");
    scanf("%d", &ch);


    switch(ch){
case 1:
//bubble sort
    for(i = 1; i < n; i++) {
        comp += 1;
        for (j = 0; j < n - i; j++){
            swap += 1;
            if( a[j] > a[j+ 1]){
                temp = a[j];
                a[j] = a[j+ 1];
                a[j+ 1] = temp;

            }

        }

    }

    printf("Sorted array: \n");
```

```
    for (i = 0; i < n; i++){
        printf("%d\t", a[i]);

    }
    printf("\n");
    break;

case 2:
//insertion sort
    for(i = 1; i < n; i++){
        index = a[i];
        for(j = i -1; j >= 0; j--){
            if ( a[j] > index){
                a[j + 1] = a[j];

            }
            else break;


        }
        a[j +1] = index;


    }

    printf("Sorted array: \n");
    for (i = 0; i < n; i++){
        printf("%d\t", a[i]);

    }
    printf("\n");
    break;


case 3:
//selection sort
    for(i = 1; i < n; i++){
        small = i - 1;
        for(j = i; j < n; j++){
            if (a[j] < a[small]){
```

```
                small = j;
            }

            if (small != i - 1){
                temp = a[small];
                a[small] = a[i - 1];
                a[i - 1] = temp;

            }


        }


    }

    printf("Sorted array: \n");
    for (i = 0; i < n; i++){
        printf("%d\t", a[i]);

    }
    printf("\n");
    break;
case 4:
    exit(0);

default:
    printf("Invalid choice.");
    break;
}
break;
}

while(ch < 4);


    return 0;
}
```

Output:

```
D:\OneDrive\Dokumen\Clg_work>cd "d:\OneDrive\Dokumen\Clg_work\Assignments\" && gcc 1sorting.c -o 1sorting && "d:\OneDrive\Dokumen\Clg_work\Assignments\"1sorting
Enter size of array: 5
Enter elements of array:
2 1 3 4 5
Unsorted array:
2       1       3       4       5
Menu:
1. Bubble sort
2. Insertion sort
3. Selection sort
4. Exit
Enter corresponding number for sorting type or exit: 1
Sorted array:
1       2       3       4       5
Enter corresponding number for sorting type or exit: 2
Sorted array:
1       2       3       4       5
Enter corresponding number for sorting type or exit: 3
Sorted array:
1       2       3       4       5
```