

## سوال ۵:

ابتدا یک فایل با پسوند c در دسکتاپ ایجاد می کنیم با نام soal5

```
alireza@alireza-virtual-machine:~/Desktop$ touch soal5.c
```

سپس کد به زبان c را داخل این فایل تایپ می کنیم.

```
soal5.c
~/Desktop
Open [v] Save [≡] - [□] [x]

1 #include <stdio.h>
2
3 void swap(int arr[],int a){
4     int temp = arr[a];
5     arr[a] = arr[a+1];
6     arr[a+1] = temp;
7 }
8
9 void bubbleSort(int arr[], int size) {
10     int i, j;
11     for (i = 0; i < size-1; i++) {
12         for (j = 0; j < size-i-1; j++) {
13             if (arr[j] > arr[j+1]) {
14                 swap(arr,j);
15             }
16         }
17     }
18 }
19
20 void calculateMiddle(int arr[], int size){
21     int middle;
22     if(size%2 == 0) {
23         middle = (arr[size/2] + arr[size/2 - 1])/2;
24     }
25     else {
26         middle = arr[size/2];
27     }
28     printf("The middle is: %d\n",middle);
29 }
30
31 int main() {
32     int n;
33     printf("Enter n: ");
34     scanf("%d", &n);
35     printf("Enter integers:\n");
36     int arr[n];
37     for(int i=0; i<n; i++) {
38         scanf("%d", &arr[i]);
39     }
40     bubbleSort(arr,n);
41     calculateMiddle(arr, n);
42     return 0;
43 }
44
```

سپس کد را کامپایل می کنیم با استفاده از کامپایلر GCC

```
alireza@alireza-virtual-machine:~/Desktop$ gcc soal5.c -o output
```

حالا فایل خروجی که نام آن output است را ران می کنیم.

```
alireza@alireza-virtual-machine:~/Desktop$ ./output
Enter n: 5
Enter integers:
1
7
3
11
5
The middle is: 5
alireza@alireza-virtual-machine:~/Desktop$
```

### توضیحات کد:

در متد main متغیر n را تعریف می کنیم که n همان عددی است به عنوان اولین ورودی گرفتیم. سپس از کاربر درخواست می کنیم که اولین ورودی را وارد کند که همان عدد n ما است، و این ورودی را داخل متغیر n میریزیم.

در ادامه یک آرایه تعریف می کنیم و باقی اعدادی که ورودی میگیریم را داخل این آرایه نگه میداریم. این ورودی ها را داخل حلقه for از کاربر میگیریم و در arr میریزیم. سپس از متد bubbleSort استفاده می کنیم برای مرتب سازی آرایه ورودی ها.

متد bubbleSort همان الگوریتم bubbleSort است که پیاده سازی شده و همچنین یک متد swap تعریف شده که مقدار دو خانه آرایه را با هم swap می کند.

بعد از مرتب سازی آرایه، حالا میانه آن حساب می شود. برای اینکار متد calculateMiddle طراحی شده که در این متد، چک می شود که اگر سائز آرایه یک عدد زوج بود، دو عدد وسط این آرایه با هم جمع می شوند و تقسیم بر دو می شوند، که حاصل می شود میانه. ولی اگر سائز آرایه فرد بود، عدد وسط این آرایه می شود میانه. در نهایت میانه بدست آمده به عنوان خروجی نمایش داده می شود.