

2.

```
9  #include <stdio.h>
10
11  int find_min(int arr[], int size);
12  int find_max(int arr[], int size);
13
14
15  int find_min(int arr[], int size) {
16      int min = arr[0];
17      for (int i = 1; i < size; i++) {
18          if (arr[i] < min) {
19              min = arr[i];
20          }
21      }
22      return min;
23  }
24
25  int find_max(int arr[], int size) {
26      int max = arr[0];
27      for (int i = 1; i < size; i++) {
```

```
27 for (int i = 1; i < size; i++) {  
28     if (arr[i] > max) {  
29         max = arr[i];  
30     }  
31 }  
32 return max;  
33 }  
34  
35 int main() {  
36     int numbers[] = {10, 3, 5, 12, -1, 7};  
37     int size = sizeof(numbers) / sizeof(numbers[0]);  
38  
39     int min = find_min(numbers, size);  
40     int max = find_max(numbers, size);  
41  
42     printf("Minimum value: %d\n", min);  
43     printf("Maximum value: %d\n", max);  
44  
45     return 0;  
46 }
```

3.

```
9  #include <stdio.h>
10
11  // Function declaration
12  void processArray(int arr[], int size);
13
14  int main() {
15      // Initialize array with arbitrary values
16      int numbers[] = {1, 2, 3, 4, 5};
17      int size = sizeof(numbers) / sizeof(numbers[0]);
18
19      // Function call
20      processArray(numbers, size);
21
22      return 0;
23  }
24
25  // Function definition
26  void processArray(int arr[], int size) {
27      // Increment each element of the array
28      for (int i = 0; i < size; ++i) {
29          ++arr[i];
30      }
31
32      // Decrement each element back
33      for (int i = 0; i < size; ++i) {
34          --arr[i];
35      }
```

```
35     }
36
37     // Check if elements are even or odd
38     for (int i = 0; i < size; ++i) {
39         if (arr[i] % 2 == 0) {
40             printf("Element is even\n");
41         } else {
42             printf("Element is odd\n");
43         }
44     }
45 }
```

input

Element is even
Element is odd
Element is even
Element is odd

...Program finished with exit code 0
Press ENTER to exit console