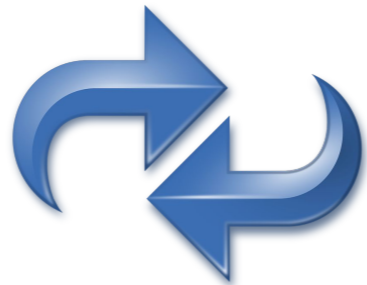


Expand an arbitrary piece of an array on the contrary (superreverse), in addition to the array and the size of the array, the function will have 2 start and finish parameters - from which number to which to unfold



```
void superReverse(int arr[], int length, int start, int finish) {
    // int reversePrint;
    int temp;
    for (int i = start; i < (start + finish) / 2; i++) {
        temp = arr[i]; // temp = 1
        arr[i] = arr[finish + start - i]; // arr[0] = - 9
        arr[finish + start - i] = temp; // last element = first element
    }
}
```

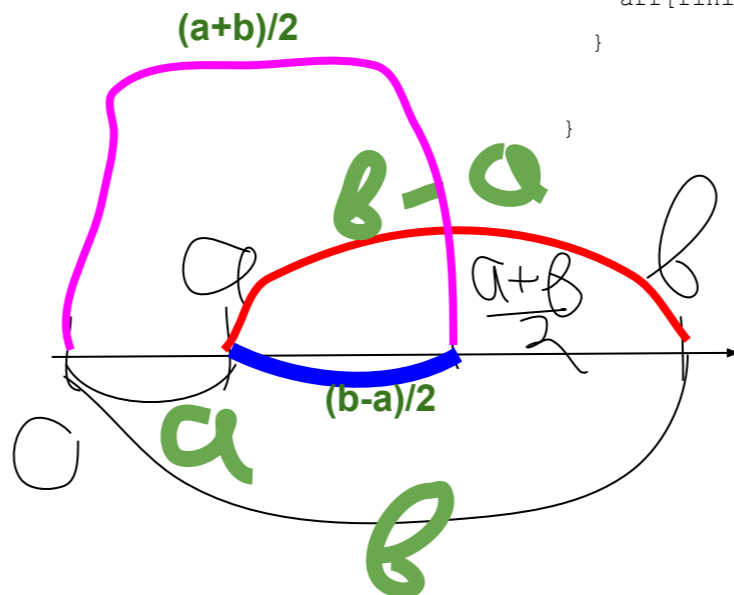
example

-6 5 2 7 9 24 234 34 2 6242 3 0 42 4 2
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

superreverse(mass,size,start,finish)

superreverse(mass,15,3,8)

-6 5 2 2 34 234 24 9 7 6242 3 0 42 4 2
 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14



```
void superReverse(int arr[], int length, int start, int finish) {
    int temp;
    while (start < finish) {
        temp = arr[start]; // temp = 1
        arr[start] = arr[finish]; // arr[0] = - 9
        arr[finish] = temp; // last element = first element
        start++;
        finish--;
    }
}
```

$$a + (b-a)/2 = 2a/2 + (b-a)/2 = (2a + b - a) / 2 = (a+b)/2$$

