Whitebox testing:

Code:- Quick Sort in C

```
#include<stdio.h>
void quicksort(int number[25],int first,int last){
   int i, j, pivot, temp;
   if(first<last){
      pivot=first;
      i=first;
      j=last;
      while(i<j){
         while(number[i]<=number[pivot]&&i<last)
         i++;
         while(number[j]>number[pivot])
         j--;
         if(i<j){
            temp=number[i];
            number[i]=number[j];
            number[j]=temp;
         }
      }
      temp=number[pivot];
      number[pivot]=number[j];
      number[j]=temp;
      quicksort(number,first,j-1);
      quicksort(number,j+1,last);
   }
}
int main(){
   int i, count, number[25];
   printf("How many elements are u going to enter?: ");
   scanf("%d",&count);
   printf("Enter %d elements: ", count);
   for(i=0;i<count;i++)
   scanf("%d",&number[i]);
   quicksort(number,0,count-1);
   printf("Order of Sorted elements: ");
   for(i=0;i<count;i++)
   printf(" %d",number[i]);
   return 0;
}
```

Test case 1:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 7
Enter 7 elements: 0
9
8
7
6
0
0
Order of Sorted elements:  0 0 0 6 7 8 9
```

Test case 2:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 6
Enter 6 elements: 7
6
5



9
8
7
Order of Sorted elements:  5 6 7 7 8 9
```

Test case 3:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 4
Enter 4 elements: null
Order of Sorted elements:  0 0 0 0
```

Test case 4:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 1
Enter 1 elements: 2
Order of Sorted elements:  2
```

Test case 5:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 6
Enter 6 elements: %
Order of Sorted elements:  0 0 0 0 0 0
```

Test case 6:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 0
Enter 0 elements: Order of Sorted elements:
```

Test case 7:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: -
Enter 0 elements: Order of Sorted elements:
```

Test case 8:

```
/tmp/qBRIgV3AVc.o
How many elements are u going to enter?: 6
Enter 6 elements: 4
5
6
7
```

A	B	C	D	E
TestCase	**Input**	**Expected Output**	**Actual Output**	**Remark**
1	0 in input - 0,9,8,7,6,0,0	0,0,0,6,7,8,9	0,0,0,6,7,8,9	Pass
2	blank spaces in input - 7,6,5, ,9,8,7	5,6,7,7,8,9	5,6,7,7,8,9	Pass
3	giving 'null' string as an input	error	0,0,0,0	Fail - no error in case of string
4	giving single number as input - 2	2	2	Pass
5	giving special character as input - %	error	0,0,0,0,0,0	Fail - no error in case of special character
6	giving '0' as an input	error	no output	Fail - should display error
7	giving '-' as number of elements	error	no output	Fail - should display error
8	giving less number of inputs than specified	should not proceed ahead	no change	Pass