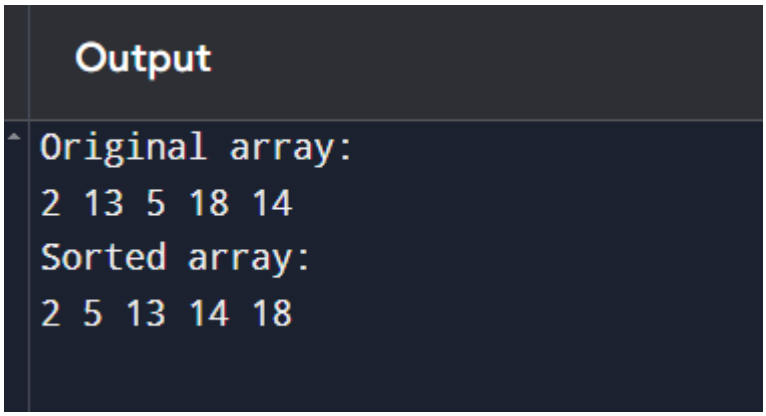
Insertion Sort

Code:

```
#include <stdio.h>
void insertionSort(int arr[], int n) {
   int i, key, j;
   for (i = 1; i < n; i++) {
      key = arr[i];
      j = i - 1;
      while (j >= 0 && arr[j] > key) {
         arr[j + 1] = arr[j];
         j = j - 1;
      }
      arr[j + 1] = key;
   }
}
void printArray(int arr[], int n) {
   for (int i = 0; i < n; i++) {
      printf("%d ", arr[i]);
   }
   printf("\n");
}
int main() {
   int arr[] = {2, 13, 5, 18, 14};
   int n = sizeof(arr) / sizeof(arr[0]);
   printf("Original array: \n");
   printArray(arr, n);
   insertionSort(arr, n);
```

```
    printf("Sorted array: \n");
    printArray(arr, n);
    return 0;
}
```

Output:



Analysis of Insertion Short:

The program implements the Insertion Sort algorithm:

- insertionSort: Sorts the array by iterating over it, shifting elements greater than the current key, and inserting the key in its correct position.
- printArray: Prints the array elements.
- main: Initializes the array, calls insertionSort, and prints the original and sorted arrays.