

## Lab 13 - wb 10/02/14

This week we are looking at a merge sort. To start off with look at the animation on wikipedia that shows how merge sort works. [http://en.wikipedia.org/wiki/Merge\\_sort](http://en.wikipedia.org/wiki/Merge_sort)

### Merge Sort

```
#include <iostream>

using namespace std;

// Merge two arrays together in order
void merge(int leftArray[], int leftSize, int rightArray[], int rightSize, int result[])
{
    // indexes for each array
    int leftIndex = 0;
    int rightIndex = 0;
    int resultIndex = 0;

    //While both arrays have stuff in them then keep checking both
    while(leftIndex < leftSize && rightIndex < rightSize)
    {
        if(leftArray[leftIndex] < rightArray[rightIndex])
            result[resultIndex++] = leftArray[leftIndex++];
        else
            result[resultIndex++] = rightArray[rightIndex++];
    }

    //If one of the arrays is empty, then merge the rest of the remaining array into the result
    while(leftIndex < leftSize)
        // copy remainder of left
        result[resultIndex++] = leftArray[leftIndex++];
    while(rightIndex < rightSize)
        // copy remainder of right
        result[resultIndex++] = rightArray[rightIndex++];
}

void mergeSort(int yourArray[], int size)
{
    int* temp = new int[size]; //temporary array to hold the result
    if (size < 2) // if the size is 1, then we cant split the array any more
        return;
    else
    {
        int mid = size / 2;

        // recursively divide the left half
        mergeSort(yourArray, mid);

        // recursively divide the right half
        mergeSort(yourArray + mid, size - mid);

        // recombine the lists in order
        merge(yourArray, mid, yourArray + mid, size - mid, temp);
    }
}
```

```

        for(int i = 0; i < size; i++)//Copy the temporary result into the array we where given
            yourArray[i] = temp[i];
    }
}

int main()
{
    const int arraySize = 9;
    //int whatToFind = 4;
    int values[arraySize] = {7,3,9,1,6,4,0,2,8};

    //Output the unsorted array
    cout<<endl<<"Unsorted Array: ";
    for(int i = 0; i < arraySize; i++)
        cout << values[i] << ", ";
    cout << endl;

    mergeSort(values, arraySize);

    //Output the sorted array
    cout<<endl<<"Sorted Array: ";
    for(int i = 0; i < arraySize; i++)
        cout << values[i] << ", ";
    cout << endl;

    cin.get();
    return 0;
}

```