```
void IterativeMergeSort(vector<int> &vect, int low, int high)
{
  vector<int> temp(high+1);
  int i, j;
  for(i=1; i<high-low; i+=i)
  {
    for(j=high-i; j>=0; j-=i+i)
    {
      IterMerge(vect, temp, max(j-i+1, 0), j, j+i);
    }
  }
}
void IterMerge(vector<int> &vect, vector<int> &temp, int low, int
mid, int high)
{
  int first=low;
  int last=mid;
  int first2=mid+1;
  int last2=high;

  int firstNew=first;

  while(first<=last && first2<=last2)
  {
    if(vect[first]<=vect[first2])
    {
      temp[firstNew]=vect[first];
      first++;
    }
    else
    {
      temp[firstNew]=vect[first2];
      first2++;
    }
    firstNew++;
  }
  while(first<=last)
  {
    temp[firstNew]=vect[first];
    first++;
    firstNew++;
  }
  while(first2<=last2)
  {
    temp[firstNew]=vect[first2];
    first2++;
    firstNew++;
  }
  for (int i=low; i<=high; i++)
  {
```

```
    vect[i]=temp[i];
  }
}
```

```
initial:
items[0] = 16
items[1] = 24
items[2] = 23
items[3] = 12
items[4] = 2
items[5] = 27
items[6] = 0
items[7] = 26
items[8] = 3
items[9] = 17
items[10] = 18
items[11] = 7
items[12] = 5
items[13] = 22
items[14] = 28
items[15] = 10
items[16] = 25
items[17] = 1
items[18] = 20
items[19] = 9
items[20] = 8
items[21] = 13
items[22] = 4
items[23] = 19
items[24] = 11
items[25] = 21
items[26] = 14
items[27] = 15
items[28] = 6
sorted:
item[0] = 0
item[1] = 1
item[2] = 2
item[3] = 3
item[4] = 4
item[5] = 5
item[6] = 6
item[7] = 7
item[8] = 8
item[9] = 9
item[10] = 10
item[11] = 11
item[12] = 12
item[13] = 13
```

```
item[14] = 14
item[15] = 15
item[16] = 16
item[17] = 17
item[18] = 18
item[19] = 19
item[20] = 20
item[21] = 21
item[22] = 22
item[23] = 23
item[24] = 24
item[25] = 25
item[26] = 26
item[27] = 27
item[28] = 28
Time (ms): 20
```