

### 算法 MPass

//一趟合并算法，该算法执行一趟合并过程，将文件 sourcelist 中长度为 length 的

//所有子文件合并到文件 mergedlist 中，n 是 sourcelist 的记录个数

void MPass(Element \*sourcelist, Element \*mergedlist, const int n, const int length)

```
{
    for( int j=1; j <=n-2*length+1; j+=2*length ) //合并相邻的两个长度为 length 的子文件
        merge(sourcelist, mergedlist, j, j+length-1, j+2*length-1);
    if( (j+length-1) < n ) //处理余留的长度小于 2*length 的子文件
        merge(sourcelist, mergedlist, j, j+length-1, n);
    else
        for( int p=j; p <=n; p++)
            mergedlist[p]=sourcelist[p];
}
```