

```
template <typename T>
void merge(fstream& f, fstream& g, fstream& h, T& dummy)
{ int n, m; T a, b;
  f.seekg(0, ios::end);
  n = f.tellg()/sizeof(T);
  g.seekg(0, ios::end);
  m = g.tellg()/sizeof(T);
  int i = 0, j = 0;
  while (i < n && j < m)
  { f.seekg(i*sizeof(T));
    f.read((char*)&a, sizeof(T));
    g.seekg(j*sizeof(T));
    g.read((char*)&b, sizeof(T));
    if (a < b)
      { h.write((const char*)&a, sizeof(T));
        i++;
      }
    else
      { h.write((const char*)&b, sizeof(T));
        j++;
      }
  }
  while (j < m)
```

```
( g.seekg(j*sizeof(T));  
  g.read((char*)&b, sizeof(T));  
  h.write((const char*)&b, sizeof(T));  
  j++;  
}  
while (i < n)  
{ f.seekg(i*sizeof(T));  
  f.read((char*)&a, sizeof(T));  
  h.write((const char*)&a, sizeof(T));  
  i++;  
}  
f.close();  
g.close();  
h.close();  
}
```