

北京交通大学 计算机实验报告

实验报告（lab6）

年 级：大学一年级
学 号：20281274
姓 名：杜海玮
专 业：计算机科学
任课老师：邹琪

二零二一年六月

目 录

1) 程序实现.....	1
A) qsort 函数.....	1
B) 对结构体数组与指针数组的比较函数实现:	1
2) 综述与学习: 模块化程序设计 (以实验六为例)	2
A) 需求分析.....	2
B) 模块化实现.....	2
3) 实验总结.....	2
A) 实验总结.....	2
B) 对小组其他成员的文档审查报告.....	3

1) 程序实现

A) qsort 函数

```
void SortArray2D(int*** input) {
    qsort(*input, Record, sizeof(*input[0]), cmp);
}
```

图 1-1 qsort 函数（以二维数组为例）

Qsort 函数有四个参数，从左到右依次是待排序目标指针 base，参与排序的目标元素个数 num，参与排序的单个元素大小 size，比较函数 comparator。本实验中因为是以三元组为单位进行排序，所以 size 是 3 个 int 的大小或如上图用 sizeof 函数进行定义。

比较函数的返回值是 int 类型，参数则是空类型的两个指针，必须进行恰当类型转换。在二维数组情况中，a 和 b 是对目标（二维数组）的指针，进行间接访问（*）后是某一行的指针，再取其第三项[2]。比较函数如下图：

```
int cmp(const void*a, const void*b) {
    int x = (*(int**)a)[2];
    int y = (*(int**)b)[2];
    return x - y;
}
```

图 1-2 cmp 函数（比较二维数组每行最后一列数字大小）

B) 对结构体数组与指针数组的比较函数实现：

```
int cmpStruct(const void* a, const void* b) {
    int x = (*(DATAITEM*)a).item3;
    int y = (*(DATAITEM*)b).item3;
    return x - y;
}

int cmpFinger(const void* a, const void* b) {
    int x = (*(DATAITEM**)a)->item3;
    int y = (*(DATAITEM**)b)->item3;
    return x - y;
}
```

图 1-3 结构体数组与指针数组的比较函数

结构体数组：

排序目标为结构体数组，则 a、b 也为结构体指针（结构体数组）类型，间接访问后直接取 item3 进行比较。

指针数组：

排序目标为指针数组，装载着指向结构体的指针，则 a、b 为结构体二重指针类型，间接访问后直接取 item3 进行比较。

```
void SortStructArray(DATAITEM** input) {
    qsort(*input, Record, sizeof(*input[0]), cmpStruct);
}

void SortFingerArray(DATAITEM*** input) {
    qsort(*input, Record, sizeof(*input[0]), cmpFinger);
}
```

图 1-4 对结构体数组与指针数组的 qsort 函数

2) 综述与学习：模块化程序设计（以实验六为例）

A) 需求分析

实验六的需求为以三元组最后一个数据为待排序值进行升序排序，分析后得出该要求不影响程序的输入及输出单元，仅需再输入后进行排序再输出即可。

B) 模块化实现

我创建 Lab6_sort.cpp 和对应头文件将排序需要的函数放入其中。于 Lab5_main.cpp 中在数据展示模块之后直接调用排序函数进行排序，之后再添加一个数据展示模块与对应介绍信息即可。在整个工程过程中原程序所属模块没有改变，仅增加了一个新功能模块，并在 main 模块中进行了整合。整个工程运行良好。

3) 实验总结

A) 实验总结

1. 实验收获:

通过本实验我实践了 qsort 函数的使用, 熟悉了其四个参数的组织形式学习编写了其对应二维数组、结构体数组和指针数组的比较函数。

通过本实验我实践了链表的一个排序进路, 冒泡排序法(交换数据域), 通过多指针配合成功的在链表这一离散数据结构中排序。但是我的方法仅交换了数据域, 仍不够完全, 需要进一步学习 qsort 函数或别的交换指针域的方法。

2. 实验困难:

对 qsort 函数的理解一度不充分, 主要是对于其第二个与第三个参数的认识: 参与排序的目标元素个数 num, 参与排序的单个元素大小 size。在二维数组中, num 就是数据记录条数, 而 size 则是整个三元组的大小(sizeof(行)或12个字节)。

在编写排序函数的时候对强制类型转化理解不清楚, 比较函数中的 a 和 b 就是两个空指针, 它应该指向待排序的元素单体, 例如在二维数组里面应该转化为(int**)型然后进行[]读取。

B) 对小组其他成员的文档审查报告

陈明强同学: qsort 函数基本相同, 排序结构体更加快速, 但缺少鲁棒性。

高原同学: qsort 函数基本相同, 函数注释上更详细。