

数据结构作业实验报告

**2018/2019(1)**



作业题目 优先级作业调度系统的模拟

学生姓名 林英琮

学生学号 201706062412

学生班级 数媒1702班

任课教师 王丽萍

提交日期 2018/12/29

**计算机科学与技术学院**

1. **作业内容**

**【问题描述】**Windows、Linux等操作系统都支持同时运行多个作业，但作业的执行顺序却因调度算法的不同而不同。通常，操作系统都采用优先级作业调度，即操作系统根据作业的长短来设置优先级大小，优先级高的作业先执行，优先级低的作业后执行。作业调度的详细情况如下描述：

一个作业Ji的长度为ti =（si，ei），si 为作业运行的开始时间（进入时间），ei 为作业运行的结束时间（离开时间），ti则为完成作业Ji所需要的执行时间（单位：秒）。作业调度的基本任务是从作业队列中选取一个来执行，如果没有作业则执行空操作操作。而优先级作业调度，是指每次选取优先级最高的作业来调度，优先级可以用优先数（每个作业一个优先数pi）来表示，优先数越小，优先级越高。作业Ji 进入系统时，即si 时刻，系统给该作业指定其初始优先数pi = ti，从而使越短的作业优先级越高。该优先数在作业等待调度执行的过程中会不断减小，调整公式为：pi = pi - wi，其中wi 为作业Ji的等待时间：wi = 当前时间-si。一旦作业被调度，该作业就一直执行，不能被抢占，只有当前执行的作业完成时，才产生下一轮调度。所以需要在每次调度前动态调整各作业的优先数。在每次调度的时候，如果出现相同优先级的作业，则按照先进先出(FIFO: First In First Out)的原则进行调度。

**【基本要求】**

1. 要求自己编程实现堆结构及其相关功能，从而实现优先级队列，**不允许使用标准模板类的堆函数和优先级队列**；测试时，各种情况都需要测试，并附上测试截图；
2. 要求采用类的设计思路，不允许出现类以外的函数定义，但允许友元函数。主函数中只能出现类的成员函数的调用，不允许出现对其它函数的调用。
3. 要求采用多文件方式：.h文件存储类的声明，.cpp文件存储类的实现，主函数main存储在另外一个单独的cpp文件中。如果采用类模板，则类的声明和实现都放在.h文件中。
4. 不强制要求采用类模板，也不要求采用可视化窗口；要求源程序中有相应注释；
5. 要求测试例子要比较详尽，各种极限情况也要考虑到，测试的输出信息要详细易懂，表明各个功能的执行正确，包括何时作业进入，何时调度哪个作业，何时离开，每个作业等待多长时间，优先数的动态变化情况等；

要求采用Visual C++ 6.0及以上版本进行调试；

1. **设计思路**

**遇到问题：**

1. 写堆的构造函数的时候：

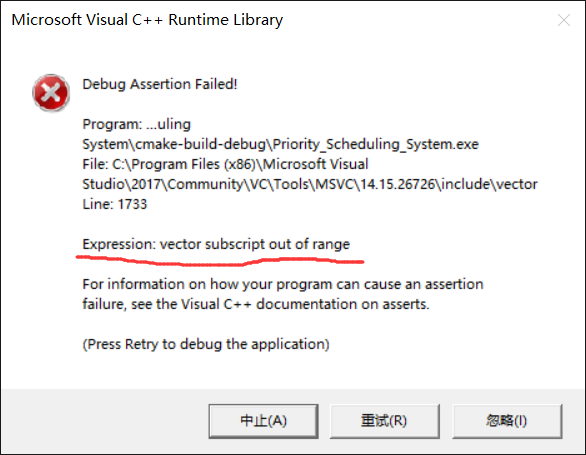
for (size\_t i = 0; i < size; i++){

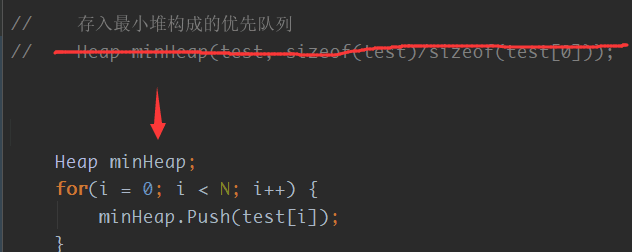
//v[i] = array[i]; //这样写不行，暂时不知道为啥。

v.push\_back(array[i]);

}

可能导致了向量v的越界。



另一种修改方式：

2.

完成最小堆的算法后，测试的过程中发现将随机生成的测试数组放入堆以后，堆仍旧无序。（改变了顺序但并未从小到大）。

解决历程：

1. 首先发现未重载swap函数，可能导致上调下调算法出错。立刻跟进
2. 随后发现Process类未重载=运算符，可能导致swap函数不正常执行。立刻跟进=的重载。

3.

1. **程序代码**

* **Process.h**

|  |
| --- |
|  |

1. **测试结果**