基于若干个需求，需要重写一个存储容器（目前采用c++内置的Stl vector容器）

1. **更快的速度**
2. **读取三元组文件构造数据结构**
3. **每个子查询的结果需要存储到Vector中，减少这个时间**
4. **最终结果的产生**
5. **更合理的内存申请分配机制、**

**目前Stl中vector的组织方式是先申请一块连续内存，当这一块内存不够用的时候，就重新申请分配一块更大的内存，然后把之前的数据拷贝过来。但是我的需求是，开辟新的空间，但是不能把之前的数据拷贝过来。就是说当申请int \*data=new int[1000]当数据超过1000的时候，应该是继续申请一个int \*data2[1000]，实现输出data[1000]的时候自动输出data2[0]，也就意味着逻辑上连接这两块内存。而不是重新拷贝。**

|  |  |
| --- | --- |
|  | Time us |
| 100 |  |
| 1000 |  |
| 10000 |  |
| 100000 |  |
| 1000000 |  |
| 10000000 |  |
| 100000000 | 内存受限 |