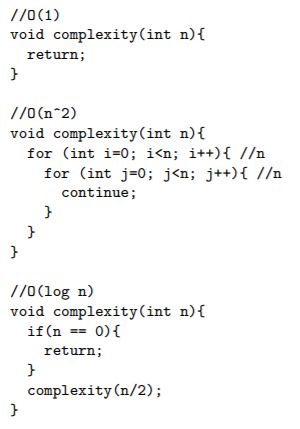
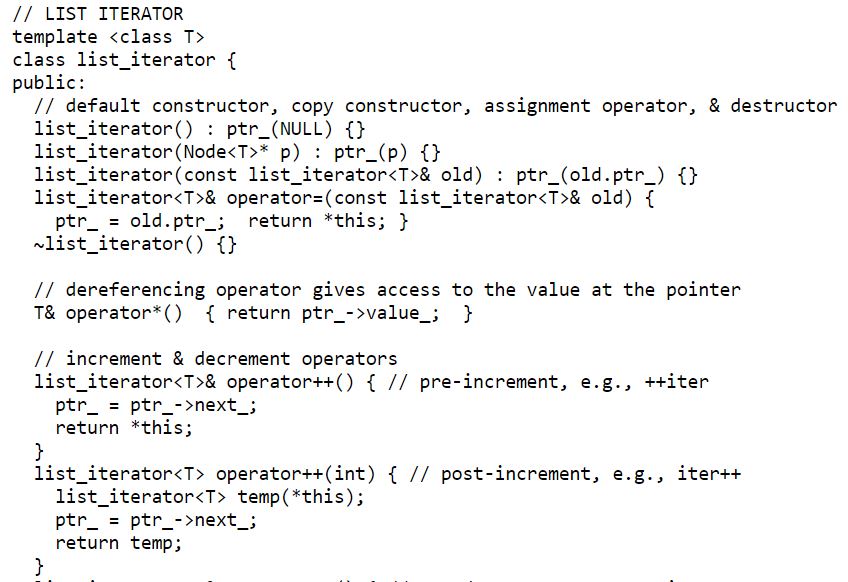
|  |  |  |  |
| --- | --- | --- | --- |
|  | Vector | Singled-linked list | STL list |
| size | √ O(1) | √ O(n) | O(1) |
| push\_back | √ O(1) | √ O(n) | O(1) |
| push\_front |  | √ O(1) | O(1) |
| pop\_back | √ O(1) | √ O(n) | O(1) |
| pop\_front |  | √ O(1) | O(1) |
| erase | √ O(n) | √ O(1) | O(1) |
| insert | √ O(n) | √ O(1) | O(1) |

|  |  |
| --- | --- |
| use of uninitialized memory | 用了没有被分配的空间 |
| mismatched new/delete/delete[] | Delete 用的不对 |
| memory leak | 没删干净 |
| already freed memory | Delete了已经被删过的东西 |
| invalid write | 书写不规范 |

|  |  |
| --- | --- |
| get a backtrace | 不知道问题在那里，去找function |
| add a breakpoint | 知道问题大概在哪里，在crash前设断点去找 |
| use step or next | 爬虫找 |
| add a watchpoint | 某一个变量在运行中变了 |
| examine different frames of the stack | recursive function的时候用 |
| use Dr Memory or Valgrind to locate the leak | Slow down  Memory problem |
| examine variable values in gdb or lldb | got an order-of-operations error or a divide-by-zero error. |



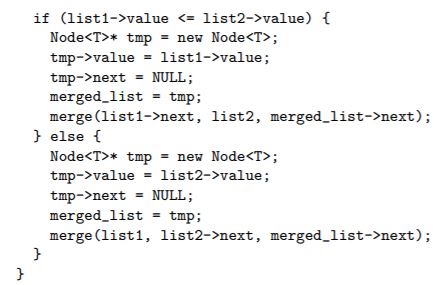
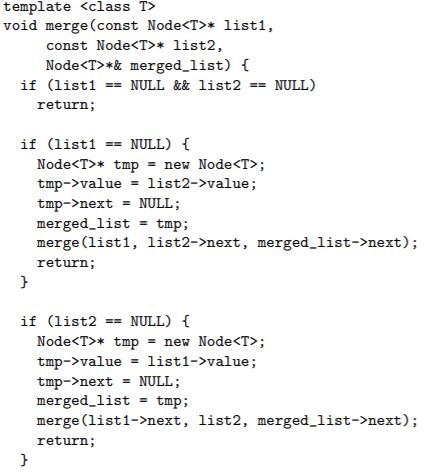
binary search：

O（log n）

merge sort:

O(n\*log n)

|  |  |  |
| --- | --- | --- |
| constructor | 构造方法 | Stairs(int s, const T& val); |
| destructor | destroy | ~Stairs(); |

merge sort：（每个Node一个数）

binary search:

template <class T>

bool binsearch(const std::vector<T> &v, int low, int high, const T &x) {

if (high == low) return x == v[low];

int mid = (low+high) / 2;

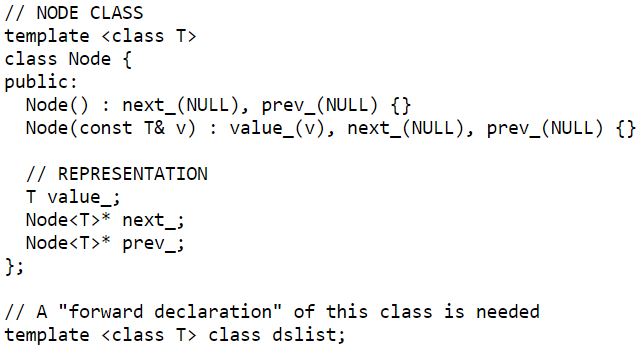
if (x <= v[mid])

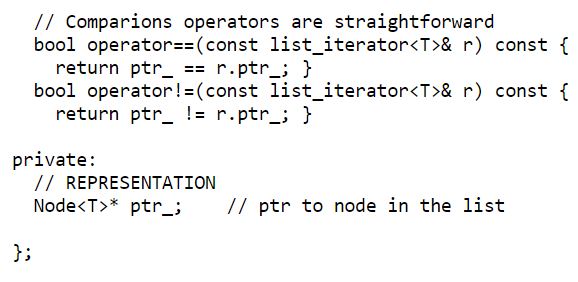
return binsearch(v, low, mid, x);

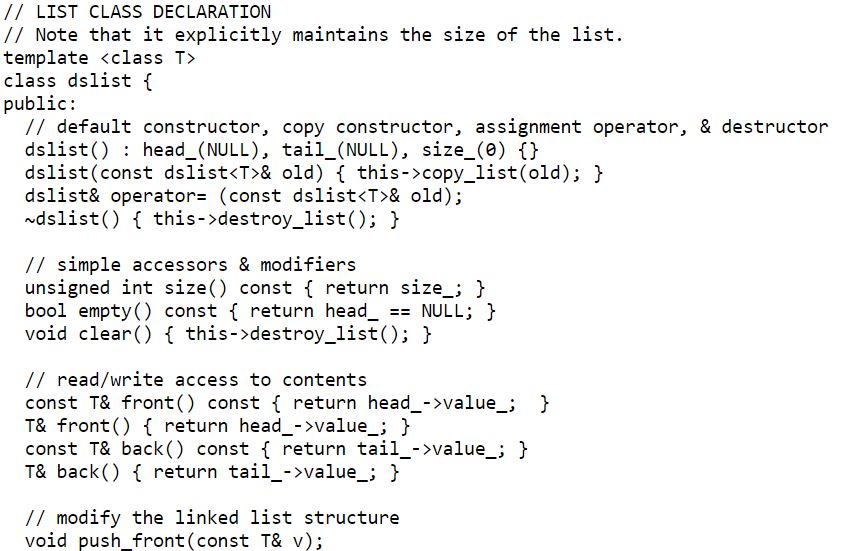
else

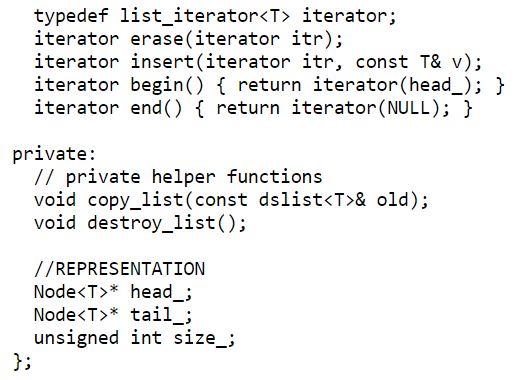
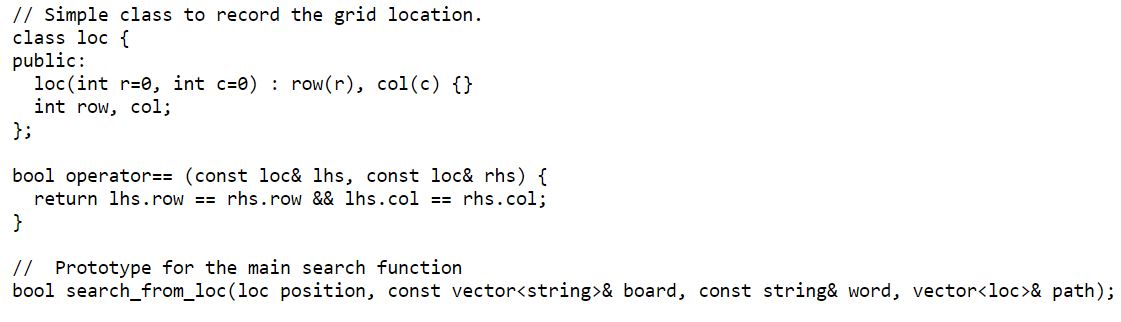
return binsearch(v, mid+1, high, x);

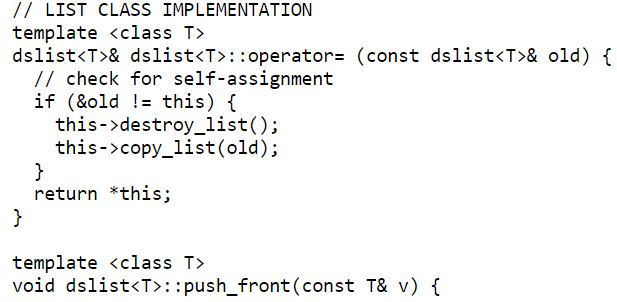
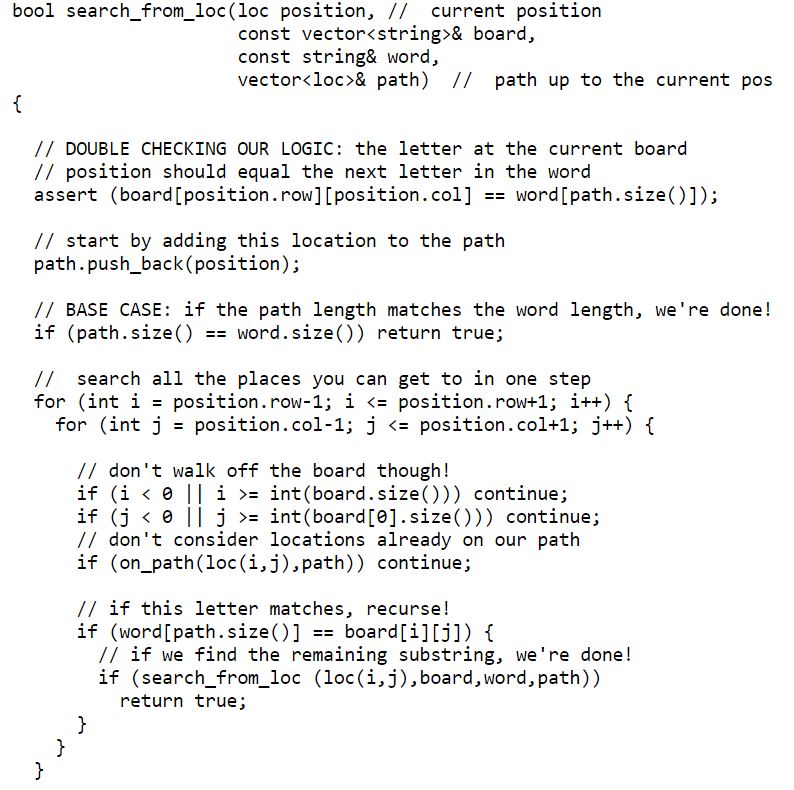
std::vector::iterator i = v.begin();

std::list::iterator p = s.begin();

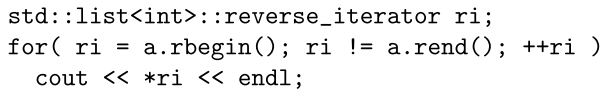












Node<T>\* p = new Node<T>();

#include <iostream> //reading & writing from keyboard

#include <cmath> //the square root function & absolute value

#include <string>//when use string, include this

#include <vector>//when use vector, include this

#include “.h” //the class head file

#include <fstream> //read and write file

cout, cin, endl, vector, list,sort前面要有std::

&：取地址运算符

\*：指针运算符（间接访问运算符）

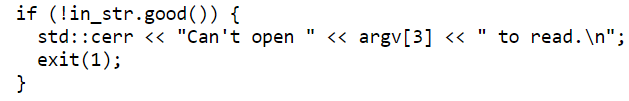
读取文件

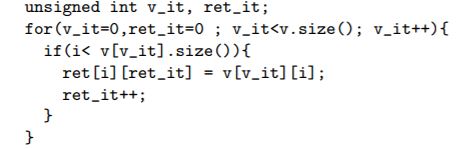
std::ifstream in\_str(argv[3]);（读取）

while (in\_str >> my\_variable) {

// do something with my\_variable

}





word search: