泛型作業四

資工二甲 張鈞奕 康智絜

說明

- 開始了新的學期
- 您當上信義和平學苑的舍監，今天有1000名學生要申請宿舍
- 請你輸入數字N，N代表宿舍住宿的缺額，電腦將會隨機挑出N名學生，並分配好房間
- 產生N筆學生資料的名單中，須包含該名學生的姓名、年級，和房號

題一

完成一個產生學生姓名及房號測資程式，學生姓名的資料請利用”學生姓名.txt”檔案來產生學生姓名的資料。輸入測資筆數，第一行印出資料筆數，接著印出學生姓名、年級及房號。

題一範例測資

Input

```
1   10
2
```

Output

```
1   10
2   Toby senior 827
3   Emanuel graduate_school 924
4   Alec graduate_school 956
5   Zayne freshman 258
6   Lewis junior 553
7   Ralph freshman 200
8   Briar graduate_school 973
9   Bridger junior 663
10  Caleb junior 555
11  Jasper sophomore 417
```

題二

完成一個產生學生姓名、年級(阿拉伯數字表達)的測資程式，學生姓名的資料請利用題一的Output檔案來產生學生姓名及隨機年級的資料。輸入測資筆數N，第一行印出資料筆數，接著N-1行印出學生姓名及年級。

題二範例測資

Input

```
10
Toby senior 827
Emanuel graduate_school 924
Alec graduate_school 956
Zayne freshman 258
Lewis junior 553
Ralph freshman 200
Briar graduate_school 973
Bridger junior 663
Caleb junior 555
Jasper sophomore 417
7
```

Output

```
7
Briar 5
Bridger 2
Zayne 1
Jasper 2
Lewis 3
Alec 1
Ralph 1
```

題三

1. 明天將是學生搬入宿舍的第一天，舍監為了方便管理住宿生，想要將學生的資料，依據樓層印出來。
(請使用vector將全部的學生資料做暫存，再將學生的資料印出低樓層到高樓層各個有人的房間的住宿生名字，依照名字大小依序列出)
2.有訪客，先輸入要找幾個人(幾筆資料)，透過名字和年級(輸入數字)印出某人住幾號房，如果找不到，印出”Not Found.”

題三範例測資

Input

```
10
Toby senior 827
Emanuel graduate_school 924
Alec graduate_school 956
Zayne freshman 258
Lewis junior 553
Ralph freshman 200
Briar graduate_school 973
Bridger junior 663
Caleb junior 555
Jasper sophomore 417
7
Briar 5
Bridger 2
Zayne 1
Jasper 2
Lewis 3
Alec 1
Ralph 1
```

Output

```
[ 1 floor ~ 2 floor ]
Room No.[200]:[ Ralph, freshman ]
Room No.[258]:[ Zayne, freshman ]

[ 3 floor ~ 4 floor ]
Room No.[417]:[ Jasper, sophomore ]

[ 5 floor ~ 6 floor ]
Room No.[553]:[ Lewis, junior ]
Room No.[555]:[ Caleb, junior ]
Room No.[663]:[ Bridger, junior ]

[ 7 floor ~ 8 floor ]
Room No.[827]:[ Toby, senior ]

[ 9 floor ]
Room No.[924]:[ Emanuel, graduate_school ]
Room No.[956]:[ Alec, graduate_school ]
Room No.[973]:[ Briar, graduate_school ]

Briar lives in Room No.[ 973 ]

Not Found.

Zayne lives in Room No.[ 258 ]

Jasper lives in Room No.[ 417 ]

Lewis lives in Room No.[ 553 ]

Not Found.

Ralph lives in Room No.[ 200 ]
```

樓層分配

1、2 樓 -> 一年級 (freshman)
3、4 樓 -> 二年級 (sophomore)
5、6 樓 -> 三年級 (junior)
7、8 樓 -> 四年級 (senior)
9 樓 -> 研究生 (graduate_school)

Class Data 中的 Definition

重新定義data object的輸入與輸出

```
 8    friend istream &operator>>(istream &input, data &d){
 9        input >> d.name >> d.grade >> d.number;
10        return input;
11    }
12
13    friend ostream &operator<<(ostream &output, data &d){
14        output << "Room No.[" << d.number <<"]:[ "<< d.name << ", " << d.grade << " ]" ;
15        return output;
16    }
```

```
18      friend bool operator<( const data& ls, const data & rs){
19          if(ls.number==rs.number)
20              return ls.name < rs.name;
21          return ls.number < rs.number;
22      }
```

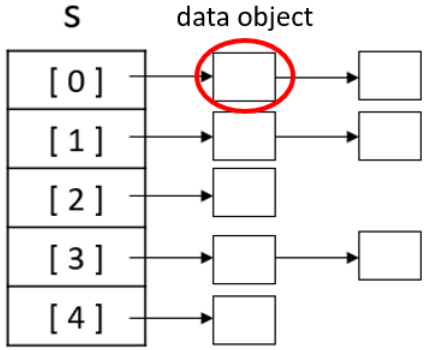
在data object存至set中時，會依據number的大小排定set中的順序。
若是number的值相同，則依據name的字母大小排順序。

```
48      vector<data>V;
49      while(cin>>cases){
50          for(int i = 0; i<cases; i++){
51              getline(cin, ignore);
52              cin >> name >> grade >> num;
53              V.push_back(data(name, grade, num));
54          }
55
```

用vector先暫存讀入的測資資料
將傳入的測資包裝成沒有名字的data class的object { data (name, grade, num) }，存入vector中

```
46      set<data>s[5];

56      for(size_t i = 0; i<V.size(); i++)
57      {
58          int tmp = V[i].getNumber();
59          switch(tmp/100)
60          {
61          case 1 : case 2 :
62              s[0].insert( V[i] );
63              break;
64          case 3 : case 4 :
65              s[1].insert( V[i] );
66              break;
67          case 5 : case 6 :
68              s[2].insert( V[i] );
69              break;
70          case 7 : case 8 :
71              s[3].insert( V[i] );
72              break;
73          case 9 :
74              s[4].insert( V[i] );
75              break;
76          default:
77              break;
78          }
79      }
```



<從vector中讀取資料存入set中>

- 依據讀入的data object中number的值來分配存入的位址
- s[i]會依據在data class中定義的operator<來排定先後順序

```
100     set<data>::iterator find_room_num;
101     find_room_num = find_if(s[j].begin(), s[j].end(), [&q_name, &q_grade](data s)->bool{
102                             string str_q_grade;
103                             switch(q_grade){
104                                 case 1 :str_q_grade = "freshman"; break;
105                                 case 2 :str_q_grade = "sophomore"; break;
106                                 case 3 :str_q_grade = "junior"; break;
107                                 case 4 :str_q_grade = "senior"; break;
108                                 default :str_q_grade = "graduate_school"; break;
109                             }
110                             return q_name == s.getName()
111                                     && str_q_grade == s.getGrade();}
112     );
```

1. 用find_if去traverse set container
2. find_if 第三個參數用Lambda expression來判斷是否找到符合的data。若Lambda 回傳 true，find_if則return該data的位址；反之，則回傳set的end()。

```
[&q_name, &q_grade](data s)->bool{
  string str_q_grade;
  switch(q_grade){
      case 1 :str_q_grade = "freshman"; break;
      case 2 :str_q_grade = "sophomore"; break;
      case 3 :str_q_grade = "junior"; break;
      case 4 :str_q_grade = "senior"; break;
      default :str_q_grade = "graduate_school"; break;
  }
  return q_name == s.getName()
          && str_q_grade == s.getGrade();}
```

[&q_name, &q_grade]是為了要取得lambda expression的外部資料

Time Complexity

存入vector的Big O

```
for(int i = 0; i<cases; i++)
{
    getline(cin, ignore);
    cin >> name >> grade >> num;
    V.push_back(data(name, grade, num));
}
```

- 若讀入n筆資料，則push_back到vector的時間是O(n)

Set 的 Data structure

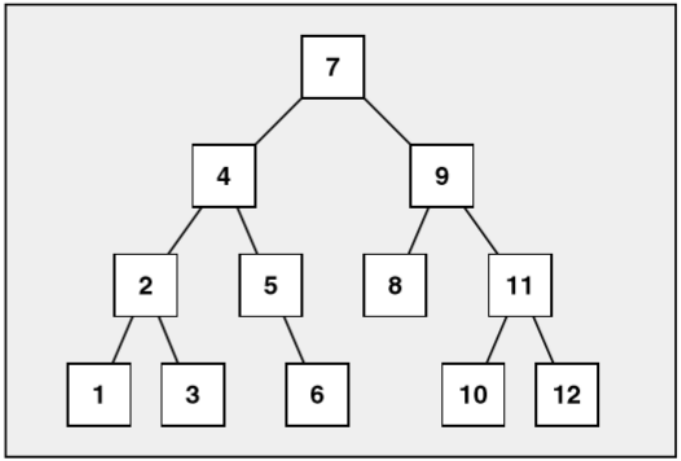


Figure 7.13. Internal Structure of Sets and Multisets

Insert 至 set 的 Big O

```
set<data>s[5];
for(size_t i = 0; i<V.size(); i++){
    int tmp = V[i].getNumber();
    switch(tmp/100){
    case 1 : case 2 :
        s[0].insert( V[i] );
        break;
    case 3 : case 4 :
        s[1].insert( V[i] );
        break;
    case 5 : case 6 :
        s[2].insert( V[i] );
        break;
    case 7 : case 8 :
        s[3].insert( V[i] );
        break;
    case 9 :
        s[4].insert( V[i] );
        break;
    default:
        break;}
}
```

- 將vector中的n筆資料依據÷100後分成五組，所以n = a + b + c + d + e
- set 因為是由紅黑樹所實作的，所以s[0~4]的Big O分別為: O(log(a)), O(log(b)), O(log(c)), O(log(d)), O(log(e))，所以insert到s的時間複雜度為O(log(a*b*c*d*e))。

Set 陣列的 find_if()

```
while(q_cases--){
    cin >> q_name >> q_grade;
    for (size_t j = 0; j < 5; ++j){
        set<data>::iterator find_room_num;
        find_room_num = find_if(s[j].begin(), s[j].end(),
                                [&q_name, &q_grade](data s)->bool{
                                    string str_q_grade;
                                    switch(q_grade){
                                        case 1 :str_q_grade = "freshman"; break;
                                        case 2 :str_q_grade = "sophomore"; break;
                                        case 3 :str_q_grade = "junior"; break;
                                        case 4 :str_q_grade = "senior"; break;
                                        default :str_q_grade = "graduate_school"; break;
                                    }
                                    return q_name == s.getName()
                                           && str_q_grade == s.getGrade();
                                }
        );
```

- 若要尋找的資料有m筆，則worst case要找 m*log(a*b*c*d*e)次，則Big O為O(m*log (a*b*c*d*e))