补充:

6、用位运算方式记录多开关的状态切换

假设一共有 A-J 共 10 个开关,相互之间独立,没有依赖关系,每个开关都仅有 ON/OFF 两种状态,初始值全部为 OFF。

- 要求:① 键盘输入 A ON / H OFF 的形式,表示某个开关置 ON/OFF
 - ② 程序要循环输入,输入错误则继续输入,输入 END 则表示结束,所有输入大小写均可
 - ③ 所有开关的状态,只能记录在一个 short 型变量中,即使用 short 型变量的 16 个 bit 中的低 10 个 bit 来表示(右起第 1bit 表示 A,第 10bit 表示 J)
 - ④ 程序执行后,输出 10 个开关的初始状态,每次改变某个开关的状态后,都输出 10 个开关的状态,以验证设置是否正确(主要看是否影响到其它开关/本质就是对某个 bit 位置 0/1 而不要影响其它 bit 位)

例: short on-off-switch = 0x0000;

依次输入: D ON, 则 short on-off-switch 为 0x0008

G ON, 则 short on-off-switch 为 0x0048

JON, 则 short on-off-switch 为 0x0248

G OFF, 则 short on-off-switch 为 0x0208

输出形式建议如下形式 (假设 short on-off-switch 的值是 0x018D):

A B C D E F G H I J

ON OFF ON ON OFF OFF OF ON ON OFF

7、考试座位随机编排

假设N个学生,M个可安排座位,要求将N个学生随机安排到M个座位中。

要求: ① N 个学生的信息用指针数组存放,最后一项为 NULL 例: const char *name[]={"张三", "李四", "王五", ..., NULL};

② M 个座位的信息用 short 型数组存放,每个数表示一行(规定最多 16 列),数的某个bit 位为 1 表示该座位可安排,0 表示不可安排;行数不限,以最后一项全 0 表示结束

例 1: 假设北楼某教室的座位排列形式如下图所示

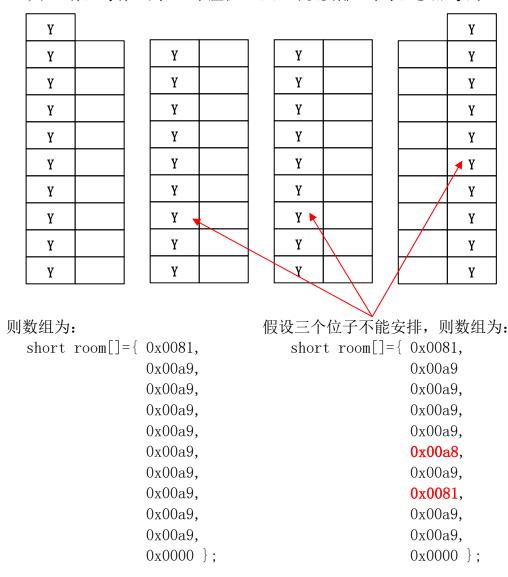
(共9行,每行9列,3个座位一组,可以安排一个学生参加考试

Y	
Y	
Y	
Y	
Y	
Y	
Y	
Y	
Y	

坐位—	组 ,	以安排	一个	字生参	加考试)
	Y				Y	
	Y				Y	
	Y				Y	
	Y				Y	
	Y				Y	
	Y				Y	
	Y				Y	
	Y				Y	
	Y		坏		Y	

```
则数组为:
                              假设一个位子坏掉,则数组为:
short room[]=\{0x0092,
                                  short room[]=\{0x0092,
               0x0092,
                                                  0x0092,
               0x0092,
                                                  0x0092,
                                                  0x0092,
               0x0092,
               0x0092,
                                                  0x0092,
               0x0092,
                                                  0x0082.
               0x0092,
                                                  0x0092,
               0x0092,
                                                  0x0092,
               0x0092,
                                                  0x0092,
               0x0000 };
                                                  0x0000 };
```

例 2: 假设南楼某教室的座位排列形式如下图所示 (共 10 行,每行 8 列,2 个座位一组,可以安排一个学生参加考试)



③ 要求 N 的值在程序中不能固定,每次执行时通过扫描 name 数组得到 N 的值, M 的值在程序中也不能固定,每次执行时通过扫描 room 数组得到 M 的值,如果学生人数 N≤座位数 M,则可安排位置随机分配,否则给出错误提示信息,退出程序

④ 输出为最后的排列,示例如下(假设为例1的座位排列):
/ 张三 / / 李四 / / 王五 /
【注:】1、要求最左侧始终有一个空列(提示:找出所有行中最左侧的1,在该列左侧输
出空列),最终整体输出形式为矩形
例: short room[]={ 0x0002, 0x0092, 0x0000}; 则输出为
/ / / / / 王五 /
2、每列宽度8个字符对齐(假设姓名最多四个汉字)
3、最左侧的空列不包含在 16 列的范围内
⑤ 排列方法有两种:
(1)、按照先行后列的顺序,从第一行开始,每行从左至右,能分配的位置依次分配,
每个位置的人员随机分配,输出时为整个教室
例:如例1的座位排列,如果只有五个学生,则第一行3人,第二行2人,最右
侧位置为空,后面再输出7个空行)
/ 赵六 / / 李四 / / 张三 /
/
(2)、能分配的位置随机分配,每个位置人员也随机分配,输出时为整个教室
例 :如例 1 的座位排列,如果只有五个学生,则可能是以下输出
/ 钱七 / / / / / /
/ / /
/ / / / /
(3)、输出前通过简单的菜单项在两种方法中进行选择即可
⑥ name/room 两个数组的数据类型和数组名 <mark>不准</mark> 变,检查时会随机替换两个数组(学生数
量/教室座位排列数量都不定),如果替换后编译出错直接视为 不通过

【作业要求:】

- 1、3月22日前网上提交本次作业
- 2、每题所占平时成绩的具体分值见网页
- 3、超过截止时间提交作业则不得分