**微型计算机原理与接口技术课设报告**

**设计名称：**扫雷

**任课老师：**唐彦老师、吕鑫老师

**小组成员：**1506010105王婉郦

1506010107赵玉娇

1506010107戴婉茹

1506010108孔艺雪

**日 期：** 二零一七年十一月

### 【需求分析】

用汇编语言编写一个扫雷游戏。用户可自行定义棋盘大小，雷数以及游戏时间。用户通过输入一个位置来进行扫雷或标雷操作。扫雷的时候，扫中雷则游戏结束，否则尽可能多的显示非雷区域及周边布雷情况。当用户在规定时间内扫完所有非雷区域或者标记正确所有的雷，则赢得游戏。超过时间即游戏失败。

**【概要设计】**

1. 输出提示信息，用户输入棋盘大小、雷的个数、游戏时间等，开始计时，游戏开始。
2. 初始化棋盘，根据用户输入的棋盘规格及雷的个数初始化棋盘，并输出一个用户操作棋盘。
3. 输出提示信息，用户选择操作类型，包括标记雷和翻开该位置。翻开时，若该位置不是雷，显示尽量多的非雷区。
4. 玩家在是预定游戏时间内翻出所有非雷区或者标记出所有雷则获胜；超出预定时间或者踩中雷则失败。

### 【详细设计】

**用户输入：**

调用DOS中断获取用户输入，若输入非法则要求重新输入，若合法则将ASCⅡ码转化为十进制数并存入内存，以便后续操作计算。

**初始化棋盘：**

1. 根据用户输入的雷的个数dim，产生dim个不同的、在棋盘规格范围内的随机数，并将初始化棋盘中对应位置置为雷的标识。
2. 随机数产生函数。利用系统时钟计数，经过操作产生随机数。
3. 统计初始化棋盘中每个位置周围八个位置的雷的总数，并将其存到对应位置。
4. 输出一个用户操作棋盘。将用户操作棋盘中的所有位置置为展示标识，并以方形输出展示。

**计时器：**

系统显示初始化棋盘后获取当前系统时间并开始计时，并在用户输入操作位置后再获取时间按小时差化为分钟差并和当前分钟数相加后再减去初始分钟数，再将总的分钟差化为秒差和当前秒数相加后再减去初始秒数获得总秒数。若用户设置的playTime大于计算所得总秒数则显示时间仍有剩余，否则将剩余时间置0，超时，游戏失败。

**标雷扫雷操作：**

1. 在界面上显示操作选择语句，其中选择1为标雷操作，选择2为扫雷操作。
2. 若选择标雷操作，从内存中获取用户所输入的位置信息，计算得到当前位置，判断当前位置是否已经标记过了。如果标记过，则重新输入标记位置；若没有标记则标记当前位置为雷，修改当前位置的符号为“#”。
3. 将当前位置信息与生成棋盘当前位置的信息进行比较，如果当前位置确实为雷，则标记雷数和成功标记上的雷数两个值同时加1；如果当前位置并不是雷，则仅仅令标记雷数加1，若所有雷均被成功标记，则玩家胜利，游戏结束。
4. 若选择扫雷操作，先与生成棋盘的当前位置信息进行比较，如果是雷，则表示踩雷，游戏结束；若不是雷，则尽可能多地显示非零区域。

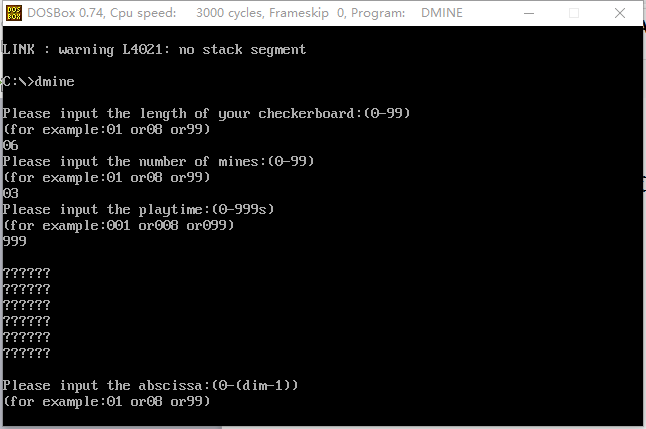
**显示非雷区域：**

当用户输入的位置选择了扫雷操作，且不为雷时，先判断当前位置位置周围八个位置是否有雷，若周围有雷，则翻开显示雷数，将数字由初始棋盘mine送至显示棋盘showMine的对应位置，且将走的步数step加一,若step等于非雷区数，则玩家获胜，游戏结束 。

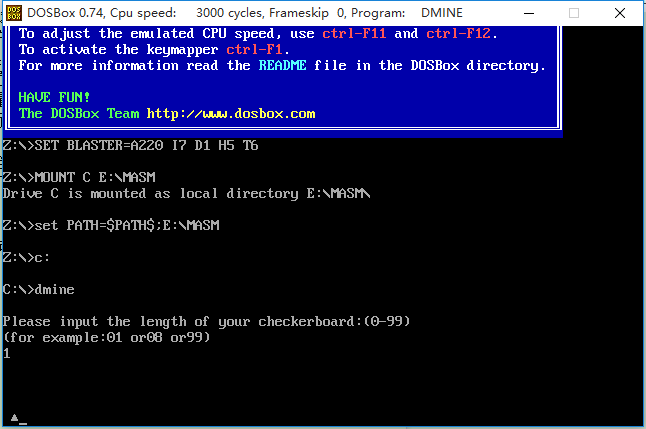
若当前位置不是雷且周围八个位置没有雷（即初始棋盘当前位置为0），则将当前位置入栈保存。并按顺序依次进入到当前位置的上、右、下、左方。先判断进入的位置是否在棋盘范围内，若不在则继续试探下一位置，若在则通过比较初始棋盘与显示棋盘同一位置的信息判断这一位置是否已经操作过，若已操作过，则转入下一位置试探；若未操作过，继续判断。先判断这一位置周围雷数是否为0，若为0，则将这一位置作为当前位置入栈保存，递归重复上一结点的相同操作；若这一位置为雷，不进行操作，转到下一位置；若不为雷，但是周围有雷，则显示雷数后s继续转入下一位置。当某一位置的上右下左四个位置都试探完毕，则将当前位置出栈，再将前一位置出栈，直到其周围位置都搜索完毕，递归遍历完所有之前经过的位置，直到到达用户输入的位置，当用户输入位置的四个方向搜索完毕，退出搜索。跳转显示显示棋盘showMine，并提示用户再次输入一个位置。

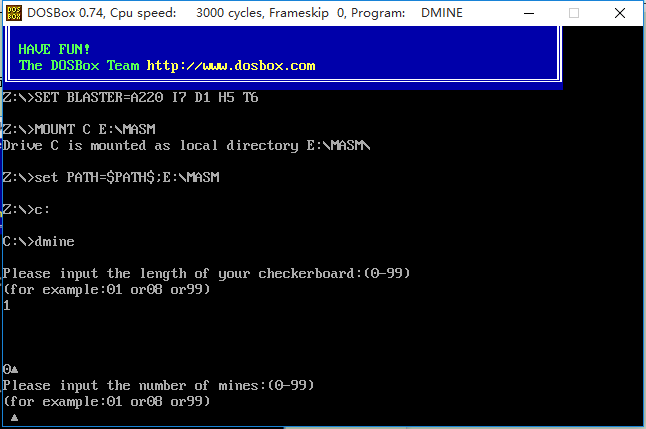
**【测试分析】**

1、用户输入棋盘长度、雷的数目及游戏时间，游戏开始，显示未翻开棋盘

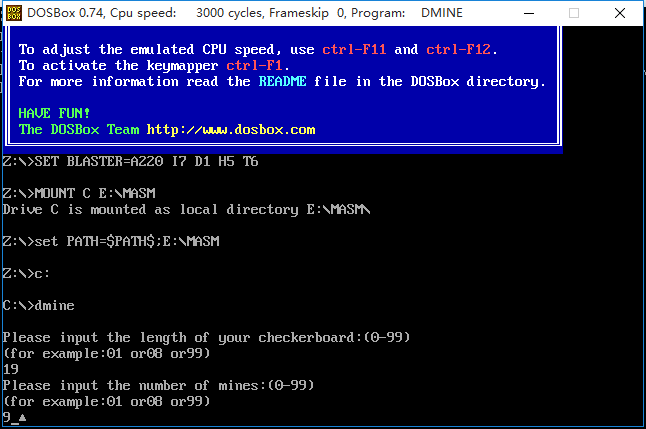


2、设置棋盘长度时，不按照规定的格式书写，程序不会结束，直到将格式完整书写；对雷数和时间长度的设置同理

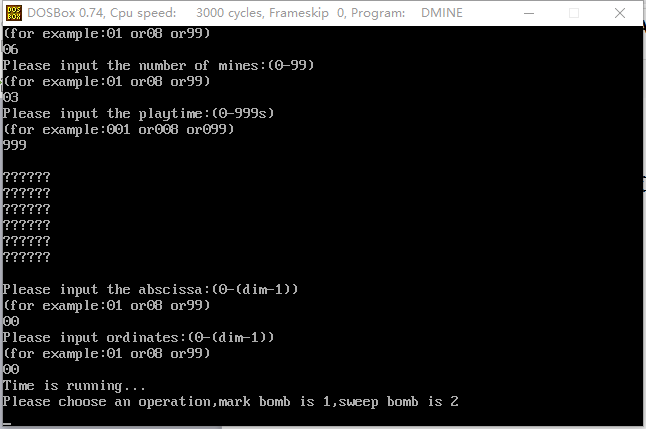




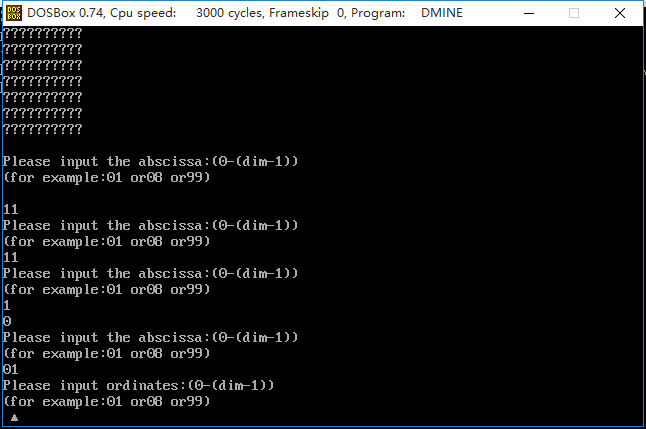
3、超过棋盘规定长度，则后输入的数字会默认设置为雷的个数；对雷数和时间长度的设置同理



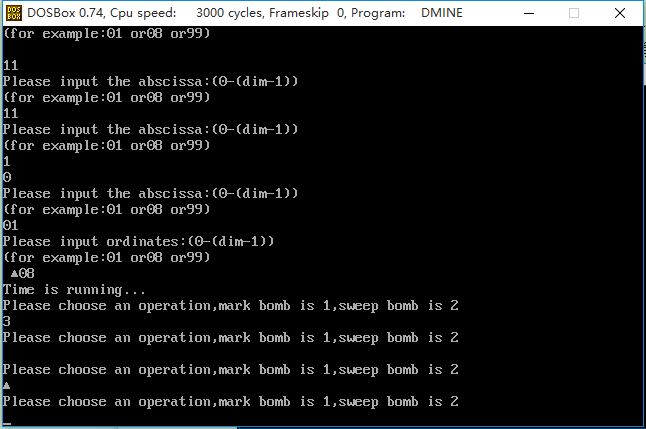
4、用户对要操作坐标进行输入并选择扫雷或标雷操作



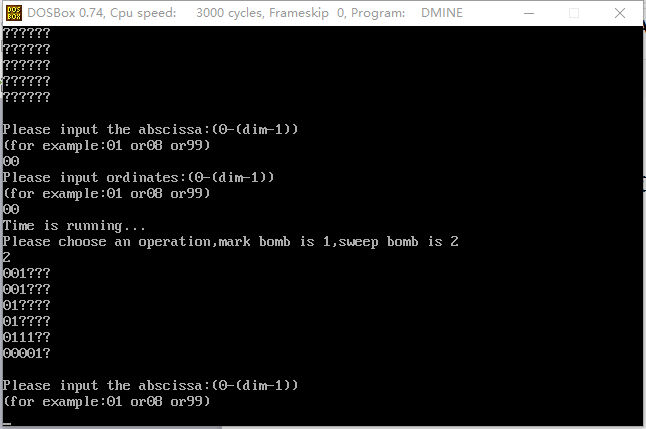
5、若横坐标的输入与格式不符合或是超过最大长度，则会跳转到提示输入横坐标的语句，重新输入，直到格式符合标准为止；纵坐标的输入同理



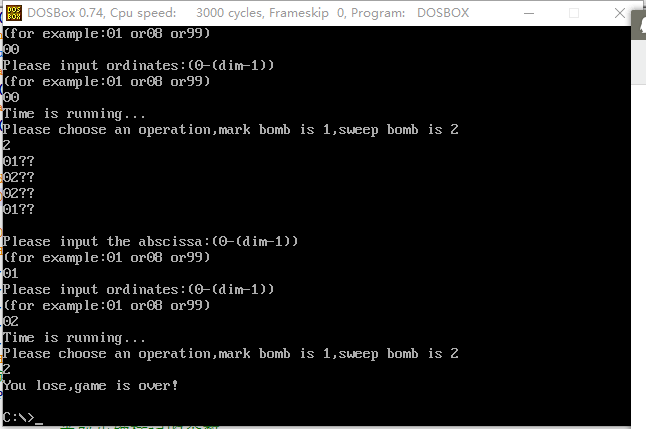
6、在选择是标雷还是扫雷操作时，输入了不存在的选项，则会跳转到操作选择语句，重新输入选项



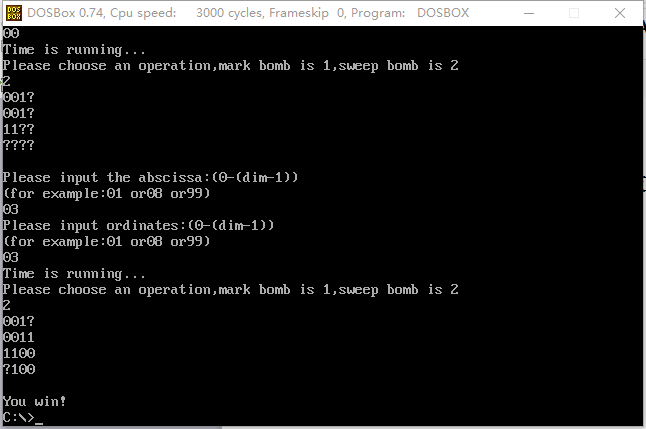
7、递归地尽可能多显示非雷区域



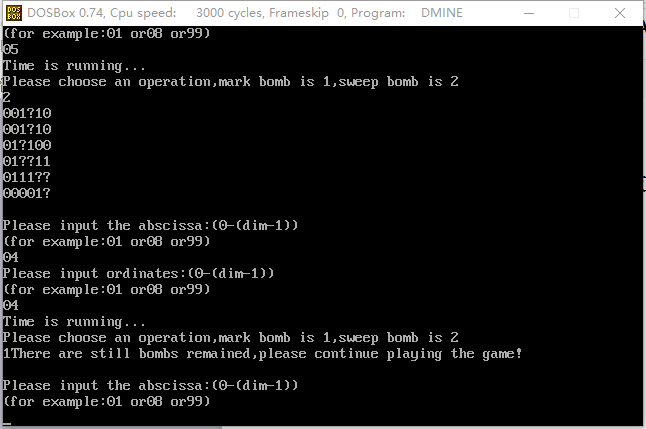
8、进行扫雷操作的位置为雷，则玩家失败游戏结束



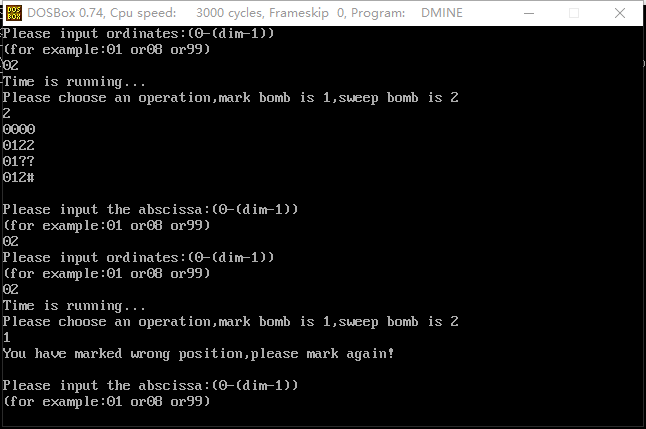
9、将所有的非雷区域翻开则获胜

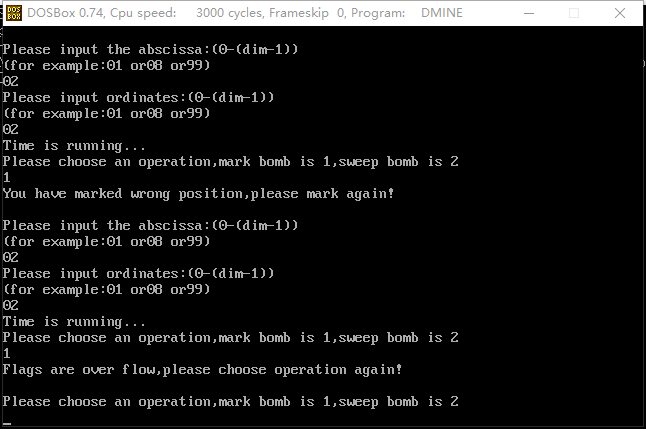
****

10、仍有雷未被标出

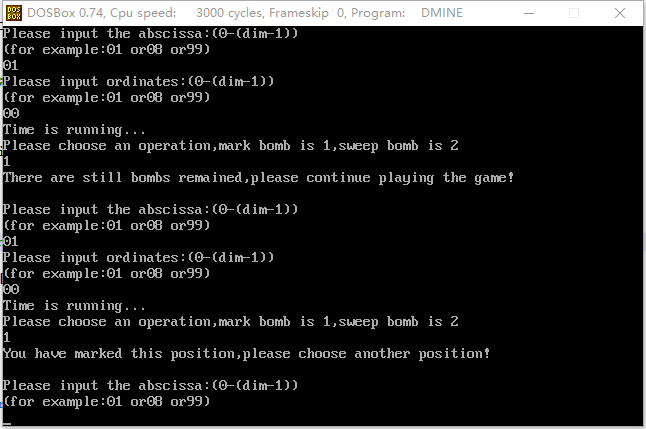


11、已标雷位置中存在标记的位置非雷

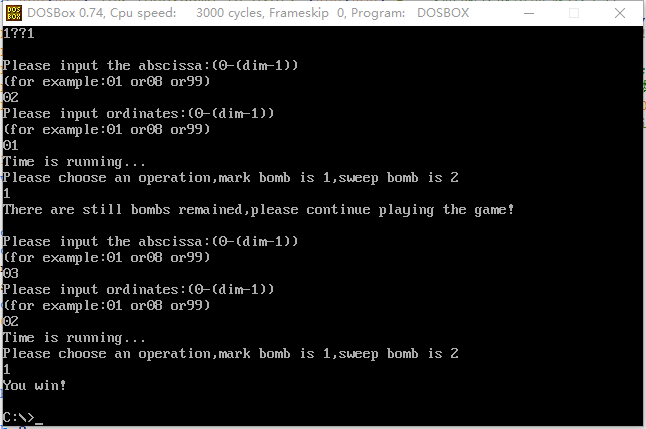
****

12、标雷的数目大于雷的实际个数，要求对当前位置重新选择操作

13、若用户已对该位置标雷，要求选择其他位置并且不会对当前位置进行操作，直接跳转到横纵坐标输入语句



14、未标记完所有的雷提示继续标记，标记完所有的雷则获胜



### **【总结】**

通过这次的汇编课设，我们收获了很多，进步了很多。首先我们已经完成了从会玩扫雷游戏到会编写扫雷游戏的转变。特别是用面向机器汇编语言来写，让我们更加清楚游戏内部逻辑的具体实现，数据在硬件中的具体存储；让我们了解了计算机硬件的工作原理，强化了我们的编程思想，提升了我们的编程能力，为我们以后学习高级语言奠定了坚实的基础。最后，非常感谢唐老师给我们这次用汇编语言编程的宝贵机会。

### 【人员分工】

棋盘初始化及棋盘显示：赵玉娇

时间设置、用户输入操作及判断游戏结束：王婉郦

扫雷操作及递归显示非雷区域：戴婉茹

用户操作判断及标雷：孔艺雪

文档编写：王婉郦、赵玉娇、戴婉茹、孔艺雪