

# 推箱子游戏

## 题目要求

1. 把推箱子游戏扩展到10个关
2. 要求可以用键盘上的上下左右箭头键控制操作及键盘上的WXAD分别代表上下左右操作键进行游戏控制操作.

## 代码改动

由于之前的代码已经把整个框架搭好了, 所以只在上面做了一些小改动.

### 1. 增加矩阵

在原来的六个矩阵的基础上, 增加了四个矩阵, 根据注释可知各个数字代表的含义.

```
// 0: 墙    1: 路    2: 人    3: 终点    4: 箱子    5: 人在终点上    6: 箱子在终点上
{
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 1, 1, 0, 0, 0, 0,
    0, 0, 0, 0, 1, 1, 0, 0, 0, 0,
    0, 0, 1, 1, 4, 4, 0, 0, 0, 0,
    0, 0, 1, 3, 1, 3, 1, 0, 0, 0,
    0, 0, 0, 0, 1, 1, 2, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, }, // 第7关
{
    0, 0, 0, 1, 1, 1, 0, 0, 0, 0,
    0, 0, 1, 1, 4, 3, 3, 0, 0, 0,
    0, 0, 4, 1, 0, 3, 4, 3, 0, 0,
    0, 1, 1, 1, 0, 3, 3, 3, 0, 0,
    0, 1, 4, 0, 0, 0, 4, 1, 0, 0,
    0, 1, 1, 4, 1, 4, 1, 0, 0, 0,
    0, 0, 2, 1, 1, 1, 1, 0, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, }, // 第8关
{
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
    0, 0, 1, 0, 0, 0, 1, 1, 0, 0,
    0, 1, 1, 1, 0, 1, 1, 1, 0, 0,
    0, 0, 1, 4, 3, 3, 4, 1, 0, 0,
    0, 2, 1, 4, 3, 6, 1, 0, 0, 0,
    0, 0, 1, 4, 3, 3, 4, 1, 0, 0,
    0, 0, 1, 1, 0, 1, 1, 1, 0, 0,
    0, 0, 0, 0, 0, 0, 0, 0, 0, 0, }, // 第9关
{
    0, 0, 0, 0, 0, 2, 0, 0, 0, 0,
    0, 0, 0, 1, 1, 1, 1, 1, 0, 0,
    0, 0, 0, 1, 0, 1, 4, 1, 0, 0,
    3, 3, 3, 4, 0, 4, 1, 0, 0, 0,
    3, 3, 3, 1, 4, 1, 1, 1, 0, 0,
    0, 0, 1, 4, 1, 0, 0, 1, 0, 0,
    0, 0, 1, 4, 1, 1, 1, 1, 0, 0,
    0, 0, 1, 1, 1, 1, 0, 0, 0, 0, } // 第10关
```

### 2. 修改相关常数, 把 countMap 、 NLEVEL 等常数进行更新

```
// 一共10关
constexpr int NLEVEL = 10;
int GameMap[8][9] = {0};
int countMap[NLEVEL] = {3,3,3,4,4,5,2,7,6,6}; // 每关的箱子/终点数
```

3. 对选择关卡部分进行更新, 因为原来的字符串缓冲区不足以写入 "10" 这个字符串.

```
//初始化消息框
case WM_INITDIALOG:
{
    // 下拉选项框
    hLevelCombo = CreateWindow("combobox", NULL, WS_CHILD | WS_VISIBLE | WS_VSCROLL | CBS_DROPDOWNLIST);
    // 添加NLEVEL个关卡数
    for (int i = 1; i <= NLEVEL; i++) {
        char temp[3] { 0 };
        _itoa_s(i, temp, 3, 10); // 10是10进制
        SendMessage(hLevelCombo, CB_ADDSTRING, 0, (LPARAM)temp);
    }
    char buff[3]{ 0 };
    _itoa_s(level + 1, buff, 3, 10);
    SendMessage(hLevelCombo, CB_SELECTSTRING, -1, (LPARAM)buff);
    return 0;
}
```

4. 增加对 `WAXD` 键的处理

在键盘按下消息中添加对这几个字母的 `case` 即可, 如下所示

```
//键盘按下消息-----
case WM_KEYDOWN:
    //用键盘操作
    switch(wParam)
    {
        case 'A':
        case VK_LEFT:    //向左键按下
            x2 = x1;
            y2 = y1 - 1;
            x3 = x1;
            y3 = y1 - 2;
            break;
        case 'D':
        case VK_RIGHT:   //向右键按下
            x2 = x1;
            y2 = y1 + 1;
            x3 = x1;
            y3 = y1 + 2;
            break;
        case 'W':
        case VK_UP:      //向上键按下
            x2 = x1 - 1;
            y2 = y1;
            x3 = x1 - 2;
            y3 = y1;
            break;
        case 'X':
        case VK_DOWN:    //向下键按下
            x2 = x1 + 1;
            y2 = y1;
            x3 = x1 + 2;
            y3 = y1;
            break;
    }
}
```