**课程设计可用到的一些参考代码:**

* **链表的定义，链表节点的创建，添加，查询，删除，显示** (参考课本)
* **清屏**： system("cls"); //注意加#include <windows.h>
* **发声**: Beep(2000,250) //第一个参数频率Hz, 第二个参数时间长短ms
* **windows 上用C语言隐藏/显示控制台光标:**

#include <Windows.h>

void HideCursor()

{

HANDLE fd = GetStdHandle(STD\_OUTPUT\_HANDLE);

CONSOLE\_CURSOR\_INFO cinfo;

cinfo.bVisible = 0;

cinfo.dwSize = 1;

SetConsoleCursorInfo(fd, &cinfo);

}

* **设置显示的位置**

#include <Windows.h>

void gotoxy(int x, int y) {

COORD pos = { x,y };

HANDLE hOut = GetStdHandle(STD\_OUTPUT\_HANDLE);// 获取标准输出设备句柄

SetConsoleCursorPosition(hOut, pos);//两个参数分别是指定哪个窗体，具体位置

}

* **控制台显示颜色;**

设置默认的控制台前景和背景颜色。

COLOR [attr]

attr----指定控制台输出的颜色属性。颜色属性由两个十六进制数字指定 -- 第一个对应于背景，第二个对应于前景。每个数字可以为以下任何值:

0 = 黑色 8 = 灰色

1 = 蓝色 9 = 淡蓝色

2 = 绿色 A = 淡绿色

3 = 浅绿色 B = 淡浅绿色

4 = 红色 C = 淡红色

5 = 紫色 D = 淡紫色

6 = 黄色 E = 淡黄色

7 = 白色 F = 亮白色

例如; color 4e

C语句: **system("color 4e");**

如果没有给定任何参数，此命令会将颜色还原到 CMD.EXE 启动时的颜色。这个值来自当前控制台窗口、/T 命令行开关或 DefaultColor 注册表值。

或使用 **SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), attr);**函数设定

* **常用文件操作**

**FILE \*fp;**

**fp=fopen(filename,"w+");** 要判断是否成功,如果fp==NULL说明打开文件失败

**fscanf(fp,"%d%s%f",&intVal, buf, &floatVal);** 从文件读取数据:

**fprintf(fp, "%d %s %.1f \n", intVal, buf, floatVal);** 数据写入到文件:

**fclose(fp);**

**Visual studio下的关键文件操作函数：**

fopen\_s( &stream, "fscanf.out", "w+" );

fprintf\_s(stream, "%s", string);

fclose(stream);

fseek( stream, 0L, SEEK\_SET );

fscanf\_s( stream, "%s", s, \_countof(s) ); // 读字符串，要加上字符串长度参数

fscanf\_s( stream, "%ld", &l ); // 读整数

fscanf\_s( stream, "%f", &fp ); // 读浮点数

fscanf\_s( stream, "%c", &c, 1 ); //读单个字符，注意要加上参数1

* **按键是否有按下,以及无回显的读取** #include <conio.h>
* **判断按键状态: if(kbhit()) 或 if(\_kbhit()) 是否有键按下**
* **无回显的读取按键函数: c=getch() 或 \_getch()**

读取键值: if (kbhit()) key=getch();

注意敲击箭头键会出现两个字符数据: LEFT(-32,75), UP(-32,72), DOWN(-32,80),Right(-32,77)

* **也可以用下列语句读取上下左右,空格键的状态**

if (GetAsyncKeyState(VK\_LEFT) ) ......左键处理语句......;

if (GetAsyncKeyState(VK\_RIGHT) )........;

if (GetAsyncKeyState(VK\_UP) )..........;

if (GetAsyncKeyState(VK\_DOWN) )........;

if (GetAsyncKeyState(VK\_SPACE)).......; //空格

* **读取系统的时间示例：**

#include <time.h>

int main()

{

**time\_t** **t**;

**struct tm** **\*lt**;

time(&t); //获取时间戳。

lt=localtime(&t);//转为时间结构。

printf("%d/%d/%d %d:%d:%d\n",lt->tm\_year+1900,lt->tm\_mon,lt->tm\_mday,lt->tm\_hour,

lt->tm\_min,lt->tm\_sec);

return 0;

}