```
#include <string.h> // for strcpy(), strcat()
#include <io.h>
#include <stdio.h>
#define Max 100000 //文件数量
int FileSearch(const char *dir);
int Write(void);
int i = -1;
typedef struct Data
{
    unsigned long size;
    char dir[500];
} Data;
Data data[Max];
int main(void)
{
    char *dir = "D:\\HL7Log\\Log"; //要读取的初始路径
    FileSearch(dir);
    printf("Total files is %d\n\n", i + 1);
    Write();
    printf("done!\n");
    return 0;
}
int Write(void) //将获得的目录栈写到本地文件
{
    FILE *fp;
    if ((fp = fopen("1.txt", "w")) == NULL)
        return -1;
    fprintf(fp, "Total files is %d\n\n", i + 1);
    while (i > -1)
    {
        fprintf(fp, "%s\t%ld\n", data[i].dir, data[i].size);
        i--;
    }
    fclose(fp);
}

int FileSearch(const char *dir) //递归遍历当前目录下的所有文件
{
    long handle;
    struct _finddata_t findData;
    char dirNew[500];
    strcpy(dirNew, dir);
    strcat(dirNew, "\\*.*");
    if ((handle = _findfirst(dirNew, &findData)) == -1L)
```

```
	{
		printf("Failed to findfrist file");
		return -1;
	}
	while (_findnext(handle, &findData) == 0)
	{
		printf("%s\n", dirNew);
		if (findData.attrib & _A_SUBDIR)
		{
			if (strcmp(findData.name, ".") == 0 || strcmp(findData.name, "..") == 0)
				continue;
			strcpy(dirNew, dir);
			strcat(dirNew, "\\");
			strcat(dirNew, findData.name);
			FileSearch(dirNew);
		}
		else
		{
			if (++i < Max) //将路径入栈
			{
				strcpy(data[i].dir, dir);
				strcat(data[i].dir, "\\");
				strcat(data[i].dir, findData.name);
				data[i].size = findData.size;
			}
		}
	}
	_findclose(handle);
}
```