

3.6 二进制读写与文件定位_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 3.6 二进制读写与文件定位涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

6.Linux 标准 io – 二进制读写与文件定位

- 在标准 I/O 中, 用于进行二进制文件进行读写时需要调用 fread 与 fwrite

函数头文件

```
#include <stdio.h>
```

函数功能

从二进制文件中读取数据到缓冲区

函数原型

```
size_t fread(void *ptr, size_t size, size_t nmemb, FILE *stream);
```

函数参数

- ptr : 缓冲区地址
- size : 读取每个数据块的大小
- nmemb : 读取数据对象的个数
- stream : 文件指针

函数返回值

- 成功 : 返回实际读取的数据对象的个数
- 失败: 当到达文件尾或者发生错误, 返回较小的数据对象个数或者 0

函数头文件

```
#include <stdio.h>
```

函数功能

将缓冲区中的数据写入到文件中

函数参数

- ptr : 缓冲区地址
- size : 读取每个数据块的大小
- nmemb : 读取数据对象的个数
- stream : 文件指针

函数返回值

- 成功：返回实际读取的数据对象的个数
- 失败：当到达文件尾或者发生错误，返回较小的数据对象个数或者 0

示例：使用 fread 与 fwrite 存储一个浮点数组的数据到文件中

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#include <time.h>
#include <errno.h>

int main(int argc, char * argv[])
{
    FILE *fp = NULL;

    fp = fopen(argv[1], "w+");
    if (fp == NULL)
    {
        fprintf(stderr, "can't open file.");
        exit(-1);
    }

    float numbers[5] = {1.1, 1.2, 1.3, 1.4, 1.5};

    size_t nwbytes = 0;
    nwbytes = fwrite(numbers, sizeof(float), 5, fp);

    if (nwbytes != 5)
    {
        fprintf(stderr, "fwrite(): %s\n", strerror(errno));
        exit(-1);
    }

    rewind(fp);

    size_t nrbytes = 0;
    float rnumbers[5] = {0.0};
    nrbytes = fread(rnumbers, sizeof(float), 5, fp);

    if (nrbytes != 5)
    {
        fprintf(stderr, "fread(): %s\n", strerror(errno));
        exit(-1);
    }

    int i;
    for (i = 0; i < 5; i++)
    {
        printf(" %f ", rnumbers[i]);
    }

    putchar('\n');

    fclose(fp);
    return 0;
}
```

- 在对文件流进行操作时，一般情况下都是顺序操作，但如果定位到文件流某一个地方进行操作，则需要使用 fseek 函数进行文件流的定位

函数头文件

```
#include <stdio.h>
```

函数功能

对文件进行定位

函数原型

```
int fseek(FILE *stream, long offset, int whence);
```

函数参数

- stream : 文件指针
- offset : 偏移量
- whence: 偏移相对位置
 - SEEK_SET : 相对于文件头
 - SEEK_CUR : 相对于文件当前位置
 - SEEK_END : 相对于文件尾

函数返回值

- 成功: 返回设置后的偏移位置
- 失败: 返回 -1, 并设置 errno

示例 : 使用 fseek 函数进行数据流的定位

```
int main(int argc, char * argv[])
{
    FILE *fp = NULL;
    char temp;
    fp = fopen(argv[1], "r");

    if (fp == NULL)
    {
        fprintf(stderr, "open failed.");
        return -1;
    }

    fseek(fp, 5, SEEK_SET);
    temp = fgetc(fp);

    printf("temp = %c\n", temp);

    fclose(fp);
    return 0;
}
~
```

练习 : 使用二进制操作接口 fread 与 fwrite 复制一张图片

全文完

本文由 简悦 SimpRead 优化, 用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta, 点击查看详细说明

