

20160314学习笔记

fputc

- `int fputc(int ch, FILE *stream);`

fgetc

- `int fgetc(FILE *stream);`

fseek

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

SEEK_SET 从文件的开始处开始搜索 这时offset要为正值

SEEK_CUR 从当前位置开始搜索

SEEK_END 从文件的结束处开始搜索 这时offset要为负值

fgets

- `char fgets(char str, int num, FILE *stream);`

fputs

- `int fputs(const char str, FILE stream);`

fread

- `int fread(void buffer, size_t size, size_t num, FILE stream);`

fwrite

- `int fwrite(const void buffer, size_t size, size_t count, FILE stream);`

```
ret = fseek(fp,-1,SEEK_CUR)//从当前位置向前偏移一个
```

```
if(0 != ret){
```

```
perror("fseek");
```

```
}
```

sscanf & sprintf

```
1.
2.  typedef struct {
3.      int num;
4.      char name[20];
5.      char sex;
6.      float score;
7.  }stu;
8.
9.  int main(int argc, char * argv[]){
10.      FILE *fp;
11.      stu s1;
12.      int ret;
13.      memset(&s1, 0, sizeof(s1));
14.      char buf[128] = {0};
15.      if(argc != 2){
16.          printf("error args\n");
17.          return -1;
18.      }
19.      fp = fopen(argv[1], "r");
20.      if(NULL == fp){
21.          perror("fopen");
22.      }
23.      if(fgets(buf, sizeof(buf), fp) == NULL){
24.          perror("fgets");
25.          return -1;
26.      }
27.      ret = sscanf(buf, "%d%s%c%f", &s1.num, s1.name, s1.sex, &s1.score);
28.      if(ret != 4){
29.          printf("ret is not 4\n");
30.          return -1;
31.      }
32.      memset(buf, 0, sizeof(buf));
33.      s1.score = s1.score + 5;
34.      ret = fprintf("buf, \"%d%s%c%f\",", s1.num, s1.name, s1.sex, s1.score);
35.      if(ret != 4){
36.          printf("ret is not 4\n");
37.          return -1;
38.      }
39.      fclose(fp);
40.
41.  }
```