

운영체제 과제 0

염선욱 물리학과 201921786

DM::OJ

PROBLEMS

SUBMISSIONS

USERS

CONTESTS

ABOUT

Hello, os201921786.

운영체제 과제 0에 대한 나의 제출

모두

Mine

Best

5 / 5 AC C	os201921786 39분 전	보기	0.01s 1.0 MB
5 / 5 AC C	os201921786 6시간 전	보기	0.02s 1.0 MB
5 / 5 AC C	os201921786 6시간 전	보기	0.02s 1.0 MB
0 / 5 WA C	os201921786 7시간 전	보기	0.02s 1.0 MB
0 / 5 WA C	os201921786 7시간 전	보기	0.01s 1.0 MB
0 / 5 RE C	os201921786 7시간 전	보기	0.02s 1.0 MB

제출 필터

현재 상태

상태 별로 필터링

언어

언어로 필터링

이동

통계

합계: 6

운영체제과제 - 89일 09:42:00

Windows 정품 인증
[설정]으로 이동하여 Windows를 정품 인증합니다.

proudly powered by DMOJ | 한국어 (ko)

DM::OJ

PROBLEMS

SUBMISSIONS

USERS

CONTESTS

ABOUT

Hello, os201921786.

os201921786의 운영체제 과제 0 제출

2024년 4월 1일 오후 3시 53분

상태 보기

원시 소스 보기

다시 제출

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct{
5     unsigned char op;
6     unsigned char len;
7 } code_tuple;
8
9 typedef struct{
10     int pid;
11     int arrival_time;
12     int code_bytes;
13 } process;
14
15
16 int main(int argc, char* argv[]){
17     FILE *fp;
18     process ps;
19     code_tuple ct;
20
21     if(argc != 2){
22         printf("Usage: cat test.bin | ./os0\n");
23         return 1;
24     }
25     if((fp = fopen(argv[1], "r")) == NULL){
26         perror("Open");
27         exit(1);
28     }
29     while(fread(&ps, sizeof(process), 1, fp) == 1){
30         fprintf(stdout, "%d %d %d\n", ps.pid, ps.arrival_time, ps.code_bytes);
31         // read code op, len
32         while(fread(&ct, sizeof(code_tuple), 1, fp) == 1){
33             fprintf(stdout, "%d %d\n", ct.op, ct.len);
34         }
35         // fseek(fp, ps.code_bytes, SEEK_CUR);
36     }
37     fclose(fp);
38     return 0;
39 }

```

운영체제과제 - 89일 09:09:17

Windows 정품 인증
[설정]으로 이동하여 Windows를 정품 인증합니다.

proudly powered by DMOJ | 한국어 (ko)

첫 제출에서는 에러가 났다. `./os0 test.bin`을 실행해서 2번째 명령행 인자를 고수준 파일 입출력을 사용하는 것으로 착각했다. 프로세스1에서 프로세스2로 넘어갈 때의 논리로 고려하지 못했다.

★

DM::OJ

PROBLEMS

SUBMISSIONS

USERS

CONTESTS

ABOUT

2024년 4월 1일 오후 4시 06분

C

os201921786의 운영체제 과제 0 제출

상태 보기

원시 소스 보기

다시 제출

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct{
5     unsigned char op;
6     unsigned char len;
7 } code_tuple;
8
9 typedef struct{
10     int pid;
11     int arrival_time;
12     int code_bytes;
13 } process;
14
15
16 int main(int argc, char* argv[]){
17     // FILE *fp;
18     process ps;
19     code_tuple ct;
20
21     // if(argc != 2){
22     //     printf("Usage: cat test.bin | ./os00\n");
23     //     return 1;
24     // }
25     // if((fp = fopen(argv[1], "r")) == NULL){
26     //     perror("Open");
27     //     exit(1);
28     // }
29     while(fread(&ps, sizeof(process), 1, stdin) == 1){
30         fprintf(stdout, "%d %d %d\n", ps.pid, ps.arrival_time, ps.code_bytes);
31         // read code: op, len
32         while(fread(&ct, sizeof(code_tuple), 1, stdin) == 1){
33             fprintf(stdout, "%d %d\n", ct.op, ct.len);
34         }
35         // fseek(fp, ps.code_bytes, SEEK_CUR);
36     }
37     // fclose(fp);
38     return 0;
39 }
40
41 }
```

운영체제과제 - 89일 09:05:39

Windows 정품 인증
[설정]으로 이동하여 Windows를 정품 인증합니다.

proudly powered by DMOJ | 한국어 (ko)

두 번째 제출에서는 틀렸다. 표준 입출력을 사용하기 위해서 stdin을 사용하긴 했지만, 여전히 프로세스1에서 프로세스2로 넘어가는 상황을 고려하지 못했다.

★

DM::OJ

PROBLEMS

SUBMISSIONS

USERS

CONTESTS

ABOUT

2024년 4월 1일 오후 4시 25분

C

os201921786의 운영체제 과제 0 제출

상태 보기

원시 소스 보기

다시 제출

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct{
5     unsigned char op;
6     unsigned char len;
7 } code_tuple;
8
9 typedef struct{
10     int pid;
11     int arrival_time;
12     int code_bytes;
13 } process;
14
15
16 int main(int argc, char* argv[]){
17     // FILE *fp;
18     process ps;
19     code_tuple ct;
20
21     // if(argc != 2){
22     //     printf("Usage: cat test.bin | ./os00\n");
23     //     return 1;
24     // }
25     // if((fp = fopen(argv[1], "r")) == NULL){
26     //     perror("Open");
27     //     exit(1);
28     // }
29     while(fread(&ps, sizeof(process), 1, stdin) > 0){
30         fprintf(stdout, "%d %d %d\n", ps.pid, ps.arrival_time, ps.code_bytes);
31         // read code: op, len
32         while(fread(&ct, sizeof(code_tuple), 1, stdin) > 0){
33             fprintf(stdout, "%d %d\n", ct.op, ct.len);
34         }
35         // fseek(fp, ps.code_bytes, SEEK_CUR);
36     }
37     // fclose(fp);
38     return 0;
39 }
40
41 }
```

운영체제과제 - 89일 09:02:37

Windows 정품 인증
[설정]으로 이동하여 Windows를 정품 인증합니다.

proudly powered by DMOJ | 한국어 (ko)

세 번째도 두 번째와 마찬가지로였다.

DM::OJ

PROBLEMS SUBMISSIONS USERS CONTESTS ABOUT

2024년 4월 1일 오후 5시 07분

os201921786의 운영체제 과제 0 제출

상태 보기
원시 소스 보기
다시 제출

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct{
5     unsigned char op;
6     unsigned char len;
7 } code_tuple;
8
9 typedef struct{
10     int pid;
11     int arrival_time;
12     int code_bytes;
13 } process;
14
15 int main(int argc, char* argv[]){
16     // FILE *fp;
17     process ps;
18     code_tuple ct;
19
20     // if(argc != 4){
21     //     printf("Usage: cat test.bin | ./os0\n");
22     //     return 1;
23     // }
24     // if((fp = fopen(argv[1], "r")) == NULL){
25     //     perror("Open");
26     //     exit(1);
27     // }
28     while(fread(&ps, sizeof(process), 1, stdin) == 1){
29         fprintf(stdout, "%d %d %d\n", ps.pid, ps.arrival_time, ps.code_bytes);
30         // read code: op, len
31         for(int i=0; i<ps.code_bytes/2; i++){
32             fread(&ct, sizeof(code_tuple), 1, stdin);
33             fprintf(stdout, "%d %d\n", ct.op, ct.len);
34         }
35         // fseek(fp, ps.code_bytes, SEEK_CUR);
36     }
37     // fclose(fp);
38     return 0;
39 }
40
41 }
```

운영체제과제 - 89일 09:01:05

Windows 정품 인증
[설정]으로 이동하여 Windows를 정품 인증합니다.

proudly powered by DMOJ | 한국어 (ko)

세 번째 제출에서 통과했지만, 32번째 줄에서 warning: ignoring return value of 'fread' declared with attribute 'warn_unused_result' [-Wunused-result] 경고문이 떴서 if문 안에서 fread()할 수 있도록 수정했다.

DM::OJ

PROBLEMS SUBMISSIONS USERS CONTESTS ABOUT

2024년 4월 1일 오후 11시 10분

os201921786의 운영체제 과제 0 제출

상태 보기
원시 소스 보기
다시 제출

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 typedef struct{
5     unsigned char op;
6     unsigned char len;
7 } code_tuple;
8
9 typedef struct{
10     int pid;
11     int arrival_time;
12     int code_bytes;
13 } process;
14
15 int main(int argc, char* argv[]){
16     process ps;
17     code_tuple ct;
18
19     while(fread(&ps, sizeof(process), 1, stdin) == 1){
20         // read ps info from standard input: pid, arr, code_bytes
21         fprintf(stdout, "%d %d %d\n", ps.pid, ps.arrival_time, ps.code_bytes);
22         // read code: op, len
23         for(int i=0; i<ps.code_bytes/2; i++){
24             if(fread(&ct, sizeof(code_tuple), 1, stdin) == 1){
25                 fprintf(stdout, "%d %d\n", ct.op, ct.len);
26             }
27         }
28     }
29     return 0;
30 }
31 }
```

운영체제과제 - 89일 08:56:24

Windows 정품 인증
[설정]으로 이동하여 Windows를 정품 인증합니다.

proudly powered by DMOJ | 한국어 (ko)

문제의 조건부터 자세히 파악하고 풀어야 좋겠다는 생각이 들었다. `cat test.bin | ./os0` 인데, 자꾸 `./os0 test.bin` 과 헷갈려서 헤맨 것 같다. 출력되는 줄이 많아서 초반에 에러가 났을 때도 내가 만든 코드의 출력 자체는 정답과 같은 줄 알았다. 텍스트 비교기까지 썼었는데, 좀 더 꼼꼼히 봐야 될 필요가 있을 것 같다.

그리고, 프로세스 간 구분해주기 위해서 프로세스의 코드를 읽을 때 꼭 코드의 길이만큼만 읽을 수 있도록 `ps.code_bytes/2` 만큼만 반복해서 읽어야 한다는 점이 중요했던 것 같다.