

Chat GPT prompt



You

정수를 랜덤하게 생성해 텍스트 파일과 이진 파일에 저장하고 출력하는 프로그램을 작성해줘.

- 상수 SIZE
 - 랜덤하게 생성되는 정수의 갯수로 값은 20
- 함수 main
 - 2개 파일의 파일 포인터는 fp1, fp2로 정의, 파일명은 사이즈 100의 문자형 배열 fname1, fname2에 저장
 - 첫번째 파일명은 rand1.txt, 두번째 파일명은 rand2.dat를 사용자로부터 입력 받음
 - 각 파일을 열때 에러가 있으면 “파일 <파일명> 을 열 수 없습니다”는 메시지 출력
 - 함수 write_files를 호출해 정수를 랜덤하게 생성하고 파일 fp1, fp2에 각각 저장
 - 함수 print_files를 호출해 파일 fp1, fp2로부터 정수를 읽어 실행 예제와 같이 출력
 - 라이브러리 함수 fseek을 이용해 각 파일 포인터를 파일의 끝으로 이동시키고 라이브러리 함수 ftell을 이용해 파일의 사이즈를 실행 예제와 같이 출력
- 함수 void write_files(FILE *fp1, FILE *fp2, int n)
 - 라이브러리 함수 srand로 랜덤 넘버 제너레이터의 시드를 셋팅
 - N개의 정수를 랜덤하게 반복해 생성하고 각 정수가 생성될 때마다 텍스트 파일 fp1과 이진 파일 fp2에 저장
- 함수 void print_files(FILE *fp1, FILE *fp2, int n)
 - 파일에서 읽은 정수를 저장하는 변수 value를 정의
 - 실행 예제와 같이 텍스트 파일 출력을 위한 헤딩 출력
 - 텍스트 파일 fp1을 N번 읽고 그때마다 읽은 정수값을 출력
 - 실행 예제와 같이 텍스트 파일 출력을 위한 헤딩 출력
 - 이진 파일 fp2를 N번 읽고 그때마다 읽은 정수값을 출력
- 실행 예제: 파일 크기를 포함해 실제 데이터는 다를 수 있음

첫번째 파일: rand1.txt

두번째 파일: rand2.dat

텍스트 파일 출력

82622156 1356139930 1399857899 1728355608 1608894334 1720472161

108303072 1331082095 1159619866 1316991337 559451330 1015096744

1120884640 981592996 664107518 1182541567 14963584 237369589

1593549844 1523666371

이진 파일 출력

82622156 1356139930 1399857899 1728355608 1608894334 1720472161

108303072 1331082095 1159619866 1316991337 559451330 1015096744

1120884640 981592996 664107518 1182541567 14963584 237369589

1593549844 1523666371

파일 크기 비교

The size of rand1.txt: 211

The size of rand2.dat: 80



ChatGPT

아래는 주어진 요구사항에 따라 작성된 C 프로그램의 예제입니다. 이 코드는 파일을 생성하고 랜덤 정수를 저장하며, 그 후에 파일을 다시 열어서 읽어오고 파일 크기를 출력합니다.

c

Copy code

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

#define SIZE 20

void write_files(FILE *fp1, FILE *fp2, int n);
void print_files(FILE *fp1, FILE *fp2, int n);

int main() {
    FILE *fp1, *fp2;
    char fname1[100], fname2[100];

    // 파일명 입력 받기
    printf("첫번째 파일명 입력 (예: rand1.txt): ");
    scanf("%s", fname1);

    printf("두번째 파일명 입력 (예: rand2.dat): ");
    scanf("%s", fname2);

    // 파일 열기
    if ((fp1 = fopen(fname1, "w")) == NULL) {
        printf("파일 %s을 열 수 없습니다.\n", fname1);
        return 1;
    }

    if ((fp2 = fopen(fname2, "wb")) == NULL) {
        printf("파일 %s을 열 수 없습니다.\n", fname2);
        fclose(fp1);
        return 1;
    }

    // 파일에 랜덤 정수 쓰기
    write_files(fp1, fp2, SIZE);

    // 파일 닫기
    fclose(fp1);
    fclose(fp2);
}
```

Chat GPT prompt

```
// 파일 읽어오고 크기 출력하기
if ((fp1 = fopen(fname1, "r")) == NULL) {
    printf("파일 %s을 열 수 없습니다.\n", fname1);
    return 1;
}

if ((fp2 = fopen(fname2, "rb")) == NULL) {
    printf("파일 %s을 열 수 없습니다.\n", fname2);
    fclose(fp1);
    return 1;
}

// 파일에서 읽어오고 크기 출력하기
print_files(fp1, fp2, SIZE);

// 파일 크기 출력
fseek(fp1, 0, SEEK_END);
fseek(fp2, 0, SEEK_END);
printf("### 파일 크기 비교 ###\n");
printf("The size of %s: %ld\n", fname1, ftell(fp1));
printf("The size of %s: %ld\n", fname2, ftell(fp2));

// 파일 닫기
fclose(fp1);
fclose(fp2);

return 0;
}

void write_files(FILE *fp1, FILE *fp2, int n) {
    srand(time(NULL));

    for (int i = 0; i < n; ++i) {
        int num = rand();
        fprintf(fp1, "%d ", num);
        fwrite(&num, sizeof(int), 1, fp2);
    }
}
```

```
void print_files(FILE *fp1, FILE *fp2, int n) {
    int value;

    // 텍스트 파일 출력 헤더
    printf("### 텍스트 파일 출력 ###\n");

    for (int i = 0; i < n; ++i) {
        fscanf(fp1, "%d", &value);
        printf("%d ", value);
    }

    // 이진 파일 출력 헤더
    printf("\n### 이진 파일 출력 ###\n");

    for (int i = 0; i < n; ++i) {
        fread(&value, sizeof(int), 1, fp2);
        printf("%d ", value);
    }

    printf("\n");
}
```

이 코드는 사용자로부터 파일명을 입력받고, 랜덤한 정수를 파일에 쓰고 읽어오며 파일 크기를 출력하는 간단한 프로그램입니다. 주의할 점은 랜덤 정수를 파일에 쓸 때 텍스트 파일에는 `fprintf`를, 이진 파일에는 `fwrite`를 사용하고, 파일을 읽을 때에도 텍스트 파일에서는 `fscanf`, 이진 파일에서는 `fread`를 사용한다는 것입니다.

Chat GPT prompt



You

```
// 파일명 입력 받기
printf("첫번째 파일명 입력 (예: rand1.txt): ");
scanf("%s", fname1);

printf("두번째 파일명 입력 (예: rand2.dat): ");
scanf("%s", fname2);
이 부분을 위의 printf에 "첫번째 파일: " 아래 printf에 "두번째 파일:"로 변경해주고
printf("\n### 이진 파일 출력 ###\n");
이 부분은 printf("\n\n### 이진 파일 출력 ###\n");으로 바뀌주고
printf("### 파일 크기 비교 ###\n");
이 부분은 printf("\n\n### 파일 크기 비교 ###\n");으로 바뀌줘
```

< 3/3 >



ChatGPT

수정된 부분을 반영하여 코드를 제공합니다.

```
c Copy code

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define SIZE 20

void write_files(FILE *fp1, FILE *fp2, int n);
void print_files(FILE *fp1, FILE *fp2, int n);

int main() {
    FILE *fp1, *fp2;
    char fname1[100], fname2[100];

    // 파일명 입력 받기
    printf("첫번째 파일: ");
    scanf("%s", fname1);

    printf("두번째 파일: ");
    scanf("%s", fname2);

    // 파일 열기
    if ((fp1 = fopen(fname1, "w")) == NULL) {
        printf("파일 %s을 열 수 없습니다.\n", fname1);
        return 1;
    }
}
```

```
if ((fp2 = fopen(fname2, "wb")) == NULL) {
    printf("파일 %s을 열 수 없습니다.\n", fname2);
    fclose(fp1);
    return 1;
}
```

```
// 파일에 랜덤 정수 쓰기
write_files(fp1, fp2, SIZE);
```

```
// 파일 닫기
fclose(fp1);
fclose(fp2);
```

```
// 파일 읽어오고 크기 출력하기
if ((fp1 = fopen(fname1, "r")) == NULL) {
    printf("파일 %s을 열 수 없습니다.\n", fname1);
    return 1;
}
```

```
if ((fp2 = fopen(fname2, "rb")) == NULL) {
    printf("파일 %s을 열 수 없습니다.\n", fname2);
    fclose(fp1);
    return 1;
}
```

```
// 파일에서 읽어오고 크기 출력하기
print_files(fp1, fp2, SIZE);
```

```
// 파일 크기 출력
fseek(fp1, 0, SEEK_END);
fseek(fp2, 0, SEEK_END);
printf("\n### 파일 크기 비교 ###\n");
printf("The size of %s: %ld\n", fname1, ftell(fp1));
printf("The size of %s: %ld\n", fname2, ftell(fp2));
```

```
// 파일 닫기
fclose(fp1);
fclose(fp2);
```

```
return 0;
}
```

Chat GPT prompt

```
void write_files(FILE *fp1, FILE *fp2, int n) {
    srand(time(NULL));

    for (int i = 0; i < n; ++i) {
        int num = rand();
        fprintf(fp1, "%d ", num);
        fwrite(&num, sizeof(int), 1, fp2);
    }
}

void print_files(FILE *fp1, FILE *fp2, int n) {
    int value;

    // 텍스트 파일 출력 헤더
    printf("\n### 텍스트 파일 출력 ###\n");

    for (int i = 0; i < n; ++i) {
        fscanf(fp1, "%d", &value);
        printf("%d ", value);
    }

    // 이진 파일 출력 헤더
    printf("\n\n### 이진 파일 출력 ###\n");

    for (int i = 0; i < n; ++i) {
        fread(&value, sizeof(int), 1, fp2);
        printf("%d ", value);
    }

    printf("\n");
}
```

위의 변경 사항에 따라서 사용자에게 입력되는 메시지와 출력되는 헤더가 수정되었습니다.