

Files I/O

Example 1: Write to a text file using fprintf()

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

int main()
{
    int num;
    FILE *fptr;
    fptr = fopen("C:\\program.txt","w");

    if(fptr == NULL)
    {
        printf("Error!");
        exit(1);
    }

    printf("Enter num: ");
    scanf("%d",&num);

    fprintf(fptr,"%d",num);
    fclose(fptr);

    return 0;
}
```

Example 2: Read from a text file using fscanf()

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

int main()
{
    int num;
    FILE *fptr;

    if ((fptr = fopen("C:\\program.txt", "r")) == NULL){
        printf("Error! opening file");
        // Program exits if the file pointer returns NULL.
        exit(1);
    }

    fscanf(fptr, "%d", &num);

    printf("Value of n=%d", num);
    fclose(fptr);

    return 0;
}
```

Example 3: Write a C program to read name and marks of n number of students from user and store them in a file.

```
#include <stdio.h>

int main()
{
    char name[50];
    int marks, i, num;

    printf("Enter number of students: ");
    scanf("%d", &num);

    FILE *fptr;
    fptr = (fopen("C:\\student.txt", "w"));
    if(fptr == NULL)
    {
        printf("Error!");
        exit(1);
    }

    for(i = 0; i < num; ++i)
    {
        printf("For student%d\nEnter name: ", i+1);
        scanf("%s", name);
        printf("Enter marks: ");
        scanf("%d", &marks);
        fprintf(fptr, "\nName: %s \nMarks=%d \n", name, marks);
    }

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

Example 4: Write a C program to read name and marks of n number of students from user and store them in a file. If the file previously exists, add the information of n students.

```
#include <stdio.h>

int main()
{
    char name[50];
    int marks, i, num;
    printf("Enter number of students: ");
    scanf("%d", &num);

    FILE *fptr;
    fptr = (fopen("C:\\student.txt", "a"));
    if(fptr == NULL)
    {
        printf("Error!");
        exit(1);
    }
    for(i = 0; i < num; ++i)
    {
        printf("For student%d\nEnter name: ", i+1);
        scanf("%s", name);
        printf("Enter marks: ");
        scanf("%d", &marks);
        fprintf(fptr, "\nName: %s \nMarks=%d \n", name, marks);
    }

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

Example 5: Write a C program to write all the members of an array of structures to a file using fwrite(). Read the array from the file and display on the screen.

```
#include <stdio.h>

struct student
{
    char name[50];
    int height;
};

int main(){
    struct student stud1[5], stud2[5];
    FILE *fptr;
    int i;

    fptr = fopen("file.txt", "wb");
    for(i = 0; i < 5; ++i)
    {
        fflush(stdin);
        printf("Enter name: ");
        gets(stud1[i].name);

        printf("Enter height: ");
        scanf("%d", &stud1[i].height);
    }

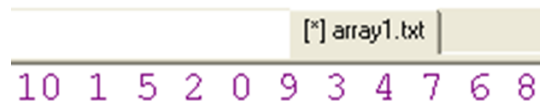
    fwrite(stud1, sizeof(stud1), 1, fptr);
    fclose(fptr);

    fptr = fopen("file.txt", "rb");
    fread(stud2, sizeof(stud2), 1, fptr);
    for(i = 0; i < 5; ++i)
```

```
{  
    printf("Name: %s\nHeight: %d", stud2[i].name, stud2[i].height);  
}  
fclose(fptr);  
}
```

BÀI TẬP THỰC HÀNH

Bài 1: Tạo một file tên **array1.txt**. Số đầu tiên trong file là số phần tử của một mảng số nguyên. Các số sau là giá trị của các phần tử theo format sau:

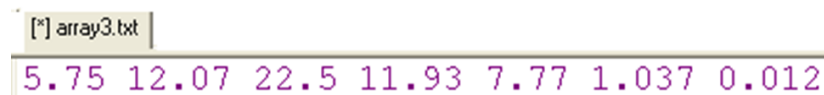


```
[*] array1.txt |  
10  
1 5 2 0 9 3 4 7 6 8
```

Viết chương trình:

- Đọc các giá trị trong file lưu vào mảng
- Xuất ra màn hình các giá trị theo thứ tự tăng dần
- Ghi kết quả câu b vào file **array2.txt** có cùng format với file array1.txt

Bài 2: : Tạo một file tên **array3.txt** chứa các số thực theo format sau:



```
[*] array3.txt |  
5.75  
12.07 22.5 11.93 7.77 1.037 0.012
```

Viết chương trình:

- In các giá trị trong file ra màn hình
- Tính trung bình các giá trị trên