Roll no. : 23112022

C program documentation extractor to streamline the process of understanding and maintaining C codebases.

Note :

- Provided program extracts content between delimeters (begins with /* and ends with */) as it is including spaces, tabs, special characters and newline i.e everything enclosed in delimeters will be considered as part of documentation.

- Even though c doesn't support nested comments, given program handles nested documentation comments also.

Program :

documentgenerator.c

```c
#include <stdio.h>

int main()
{
        FILE* fp;
        char fname[25], c, nextc;
        int cl = 0;

        printf("\nC code documentation extractor");
        printf("\n\nEnter C file name to extract documentation : ");
        scanf("%s", fname);
        printf("\n_____\n\n");

        fp = fopen(fname, "r");

        if (fp == NULL)
        {
                printf("Error: Unable to open %s file\n\n", fname);
                return 0;
        }

        printf("C program Documentation : \n");
        printf("\n-----\n");

        while ((c = fgetc(fp)) != EOF)
        {
                if (c == '/' && (nextc = fgetc(fp)) == '*')
                {
                        cl++;
                        printf("\n");
                }
                else if (c == '*' && (nextc = fgetc(fp)) == '/')
                        cl--;

                else if (cl > 0)
                        printf("%c", c);
        }

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

input file :

Open

factorial.c
~/Desktop/sem III/System-2/DocumentGenerator

```c
#include <stdio.h>

/*
Factorial function : to calculate the factorial of a number.
function uses recursion to multiply the current number (n)
by the factorial of (n-1) until it reaches the base case.
Note: This function does not handle negative numbers.
*/

int factorial(int n)
{
    /*Base case: If n is 0 or 1, return 1, Recursion stops when n is 1 or below.
    */
        if (n <= 1)
        {
                return 1;
        }
        else
        {
                /*Recursive case: Multiply n by factorial(n-1), recursion continues until the base case is reached.
                */
                return n * factorial(n - 1);
    }
}

/*Main function to demonstrate factorial function, It takes input then calculates and prints the factorial.
 /*Note: Ensure the factorial function works for all positive integers only. */
 /*Avoid passing negative values as input, as it is not handled. */
*/

int main()
{
        int num;

        printf("\nEnter number to calculate factorial : ");
        scanf("%d",&num);

        int result = factorial(num);

        printf("Factorial of %d is %d\n", num, result);

        return 0;
}
```

Output :

shruti29@inspiron: ~/Desktop/sem III/System-2/DocumentGenerator

```
shruti29@inspiron:~/Desktop/sem III/System-2/DocumentGenerator$ gcc documentgenerator.c
shruti29@inspiron:~/Desktop/sem III/System-2/DocumentGenerator$ ./a.out

C code documentation extractor

Enter C file name to extract documentation : factorial.c

_____________________________________________________________________________

C program Documentation :

-----------------------------------------------------------------------------


Factorial function : to calculate the factorial of a number.
function uses recursion to multiply the current number (n)
by the factorial of (n-1) until it reaches the base case.
Note: This function does not handle negative numbers.

Base case: If n is 0 or 1, return 1, Recursion stops when n is 1 or below.

Recursive case: Multiply n by factorial(n-1), recursion continues until the base case is reached.

Main function to demonstrate factorial function, It takes input then calculates and prints the factorial.

Note: Ensure the factorial function works for all positive integers only.

Avoid passing negative values as input, as it is not handled.

shruti29@inspiron:~/Desktop/sem III/System-2/DocumentGenerator$
```