

```

#include <stdio.h>
#include "string.h"

int DeleteComments(char *str, int Flag); //Функция удаления комментариев
const char* InputFile = "InputFile.c";
const char* OutputFile = "OutputFile.c";

int main()
{
    int Flag = 0;
    char s[256] = { '\0' };
    int i;

    FILE *in = fopen(InputFile, "rt");
    if (in == NULL) //Если файл не найден
    {
        printf("file not found\n");
        return -1;
    }
    FILE *out = fopen(OutputFile, "wt");
    if (out == NULL) //Если файл не найден
    {
        printf("Recording file not found\n");
        return -2;
    }
    while (fgets(s, 256, in) != NULL)
    {
        Flag = DeleteComments(s, Flag);
        if ((s[0] == '\0') && (Flag == 1))
            continue;
        //Если строка пустая и код возврата функции 1, то сразу
        переходим на следующую строку
        if (fputs(s, out) == EOF)
        {
            printf("Error of writing\n");
            fclose(in);
            fclose(out);
            remove(OutputFile);
            return -3;
        }
    }
    //void *memset(void *s, int c, size_t n) – размещает символ c в
    первых n позициях строки s и возвращает s
    memset(s, '\0', 256);
}

```

```

        for (i = 0; i <= 255; i++)
        {
            if(s[i] != '\0')
            {
                printf("Error of programm");
                fclose(in);
                fclose(out);
                remove(OutputFile);
                return -4;
            }
        }
    }
    if (!feof(in))//В случае ошибки считывания
    {
        printf("error of reading\n");
        fclose(in);
        fclose(out);
        remove(OutputFile);//Удаление файла
        return 0;
    }
    else
    {
        fclose(in);
        fclose(out);
    }
    printf("Success\n");
    return 0;
}

int DeleteComments(char *str, int Flag) //Функция удаления комментариев
{
    char *first;
    char *second;
    char *third;
    unsigned long Length;
    Length = strlen(str);

    if (Flag)
    {
        /*char *strstr(const char *cs, const char *ct)
        – возвращает указатель на первое вхождение ct в cs или,
        если такового не оказалось, NULL*/
        third = strstr(str, "*/");
        str[0] = '\0';
        if (third == NULL) return 1;
    }
}

```

```

else
{
    //char *strcat(char *s ,const char *ct) – приписывает ct к s;
    //возвращает s
    strcat(str, third + 2);
    if (str[0] == '\n') str[0] = '\0';
    return 0;
}
}
while ((strstr(str, "//") != NULL) || (strstr(str, "/*") != NULL))
{
    first = strstr(str, "//");
    second = strstr(str, "/*");
    if (((second == NULL) && (first != NULL)) || ((first != NULL) &&
        (second != NULL) && ((Length - strlen(first)) < (Length -
            strlen(second)))))
    {
        first[0] = '\0';
        if (str[Length - 1] == '\n') strcat(str, "\n");
        break;
    }
    third = strstr(str, "*/");
    second[0] = '\0';
    if (third == NULL) Flag = 1;
    else
        strcat(str, third + 2);
}
return Flag;
}

```