

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра Вычислительной техники

ОТЧЕТ
по лабораторной работе № 11
по дисциплине «Программирование»
ТЕМА: Битовые поля в структурах

Студент гр. 3311

Баймухамедов Р.Р

Преподаватель

Хахаев И.А.

Санкт-Петербург

2024

Цель работы

Целью работы является изучение и освоение работы битовыми полями в структурах

Задание (вариант 4)

Аппаратный адрес сетевого интерфейса (MAC-адрес) состоит из 12 шестнадцатеричных чисел, сгруппированных в 6 групп (октетов), например, fa:36:2a:c6:f8:4f (каждая группа — байт в двоичном представлении) (см. <https://ru.wikipedia.org/wiki/MAC-адрес>). Первые три октета являются уникальным идентификатором производителя (OID) в соответствии со стандартом IEEE 802-2001 (см. <http://standards.ieee.org/findstds/standard/802b-2004.html>).

Разработать алгоритм и реализовать функцию преобразования кода производителя сетевой карты с использованием битовых полей в структурах и битовых операций. Список кодов производителей - <http://standards-oui.ieee.org/cid/cid.txt>

Адреса вводятся с клавиатуры.

Постановка задачи и описание решения

Для выполнения данной лабораторной работы необходимо разработать программу, которая будет читать данные из файла, содержащий название производителя и уникальный идентификатор организации (УИО), запрашивать ввод MAC-адреса, заменять по желанию пользователя производителя MAC-адреса (УИО) на один из данных файла и выводить MAC-адрес.

Для достижения этой цели необходимо выполнить следующие шаги:

1. Посчитать количество строк в файле
2. Сформировать массив, заполненный данными, прочитанными из файла
3. Запросить у пользователя MAC-адрес

4. Запросить у пользователя номер производителя, чьим УИО будет заменен текущий УИО
5. Вывести измененный MAC-адрес

Описание функций

№	Функция	Тип	Назначение
1	main	int	Запускает программу, обрабатывает пользовательский ввод, вызывает другие функции в зависимости от выбора пользователя, и завершает программу при необходимости.
2	output_mac	void	Вывод MAC-адреса
3	print_head	void	Вывод оглавления данных
4	output_array	void	Вывод массива структуры типа MNF
5	clear_screen	void	Очистка экрана
6	input_mac	void	Ввод MAC-адреса
7	count_of_lines	int	Подсчёт количества строк в файле
8	split_string	void	Разделяет строку по символу-разделителю и записывает элементы строки в массив
9	add_to_array	void	Чтение из файла и запись элементов строк в предназначенный массив, имеющий структуру MNF
10	change_manuf	void	Смена производителя MAC-адреса

Описание переменных

int main()

№	Переменная	Тип	Назначение
1	i	int	Количество строк в файле
2	option	int	Опция, выбираемая пользователем
3	**manuf	MNF (struct manuf)	Массив производителей структуры типа MNF
4	address	MAC (struct MACAddress)	MAC-адрес

struct MACAddress

№	Название	Тип	Назначение
1	octet1	unsigned:8	Первый октет MAC-адреса

2	octet2	unsigned:8	Второй октет MAC-адреса
3	octet3	unsigned:8	Третий октет MAC-адреса
4	octet4	unsigned:8	Четвёртый октет MAC-адреса
5	octet5	unsigned:8	Пятый октет MAC-адреса
6	octet6	unsigned:8	Шестой октет MAC-адреса

struct manuf

№	Название	Тип	Назначение
1	*name	char	Название производителя
2	octet1	unsigned:8	Первый октет УИО производителя
3	octet2	unsigned:8	Второй октет УИО производителя
4	octet3	unsigned:8	Третий октет УИО производителя

void output_array(MNF *ARR[], int n)

№	Переменная	Тип	Назначение
1	i	int	Количество строк массива

void input_mac(MAC *address)

№	Переменная	Тип	Назначение
1	input[MAXLEN]	char	Вводимый пользователем MAC-адрес в виде строки
2	a	unsigned int	Октет MAC-адреса в 16-ричной системе исчисления
3	i	int	Номер октета
4	*octet[6]	char	Хранение октетов
5	wordCount	int	Кол-во элементов, на которые строка была разделена

int count_of_lines(char *filename)

№	Переменная	Тип	Назначение
1	line[maxlen]	char	Хранение информации строки из файла
2	n	int	Счётчик количества строк в файле

void split_string(char *inputString, char **words, int *wordCount, char delimiter)

№	Переменная	Тип	Назначение
1	wordIndex	char	Индекс текущего слова в массиве words
2	wordStart	char	Индекс начала текущего слова в строке inputString
3	wordLength	char	Длина текущего слова
4	inWord	int	Флаг, указывающий на то, находится ли функция внутри слова или не внутри
5	i	int	Переменная для итерации по символам в строке inputString.
6	len	int	Длина элемента в строке

void add_to_array(MNF *ARR[], char *filename, int n, char sep)

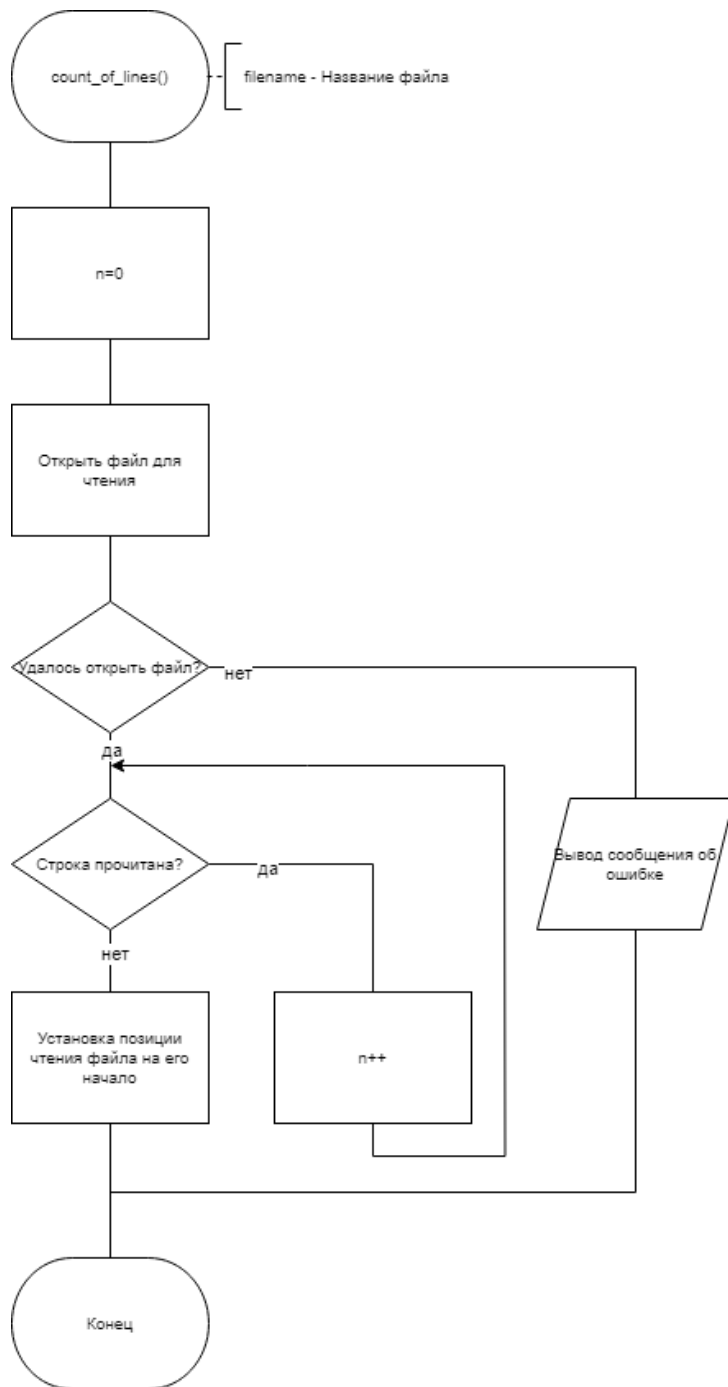
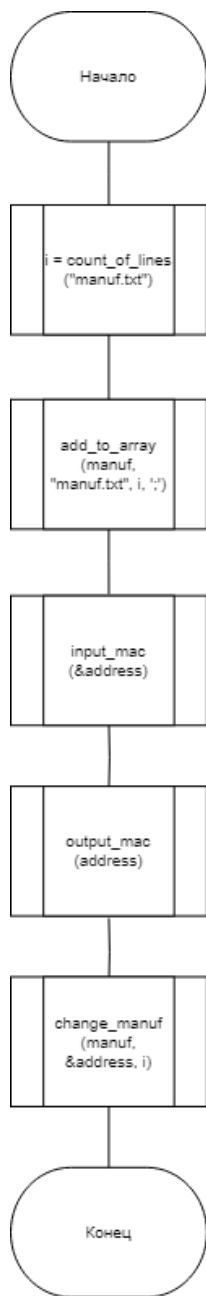
№	Переменная	Тип	Назначение
1	**TMP	char	Временный массив, куда записываются данные из строки файла
2	line[MAXLEN]	char	Хранение информации строки из файла
3	wordCount	int	Кол-во элементов, на которые строка была разделена

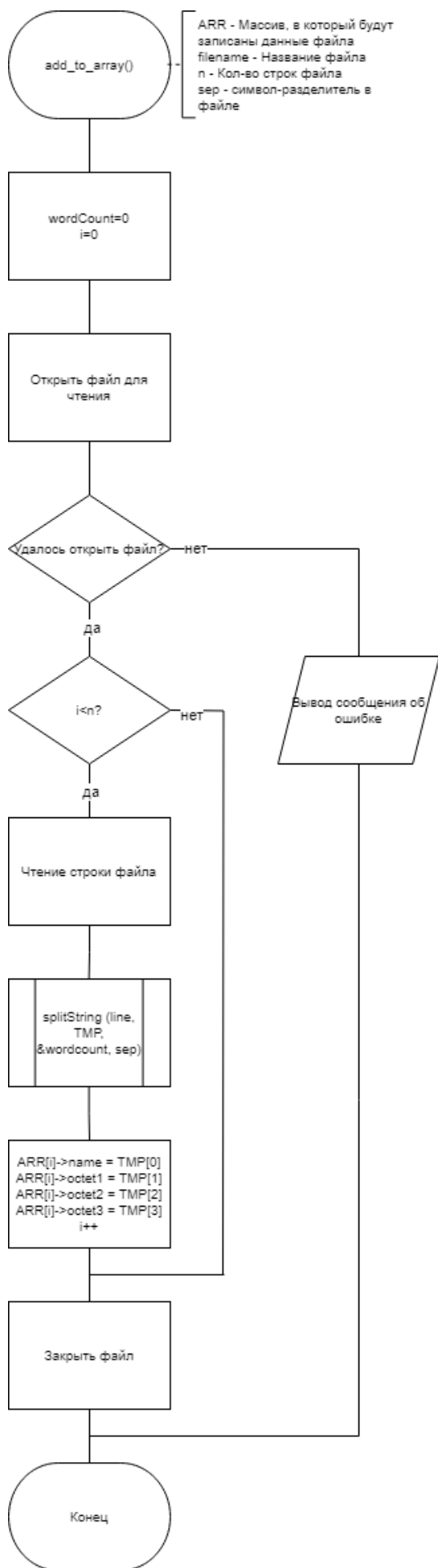
4	i	int	Переменная в цикле, отвечающая за номер элемента массива
5	j	int	Переменная в цикле, отвечающая за номер элемента строки
6	temp_octet1	unsigned int	Хранение значения первого октета MAC-адреса
7	temp_octet2	unsigned int	Хранение значения второго октета MAC-адреса
8	temp_octet3	unsigned int	Хранение значения третьего октета MAC-адреса

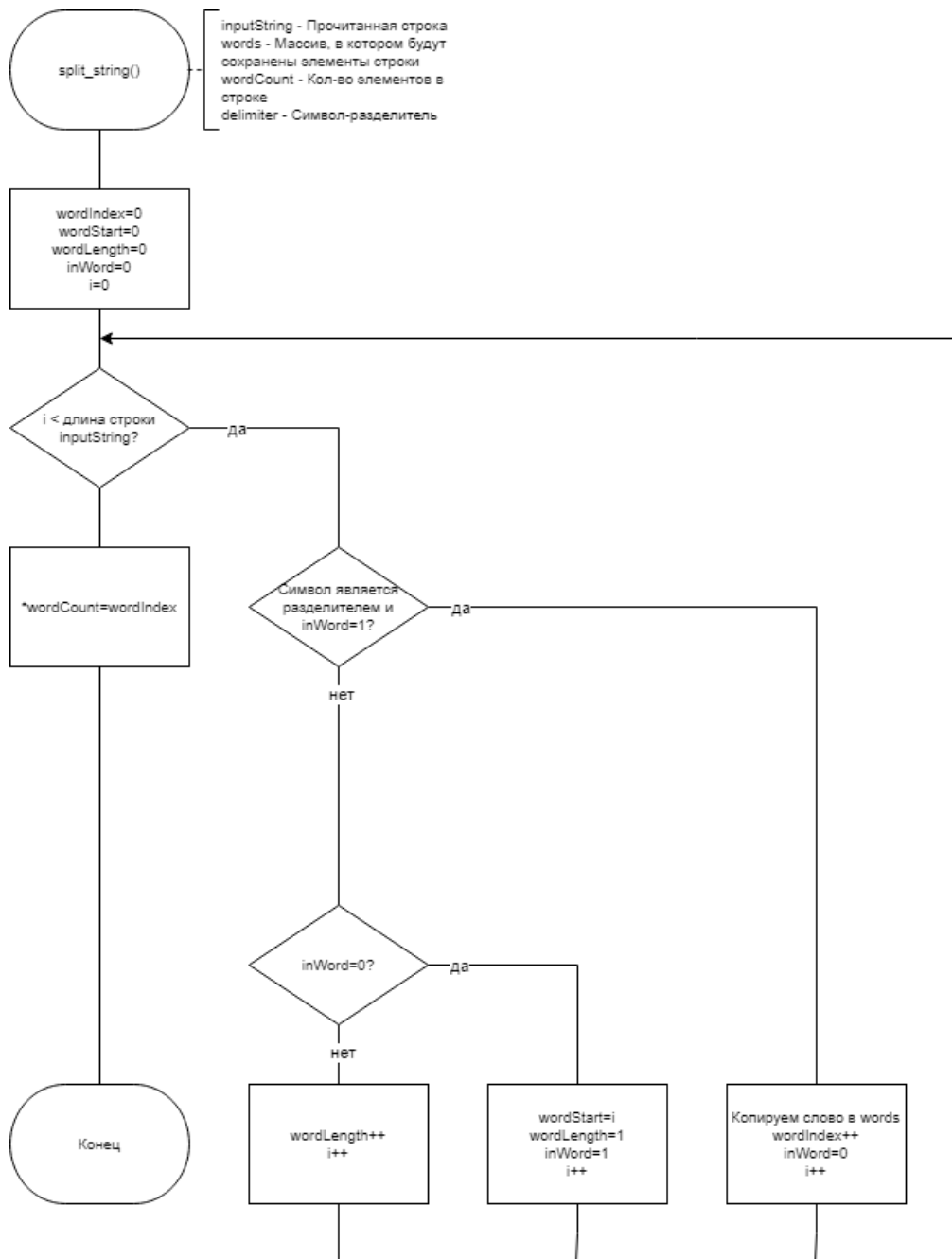
void change_manuf(MNF *ARR[], MAC *address, int i)

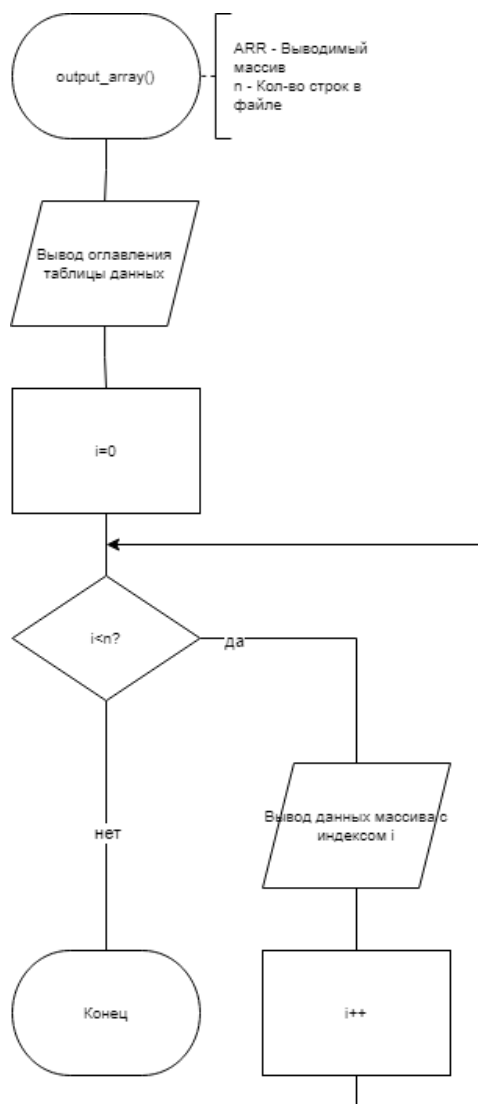
№	Переменная	Тип	Назначение
1	number	unsigned int	Вводимый пользователем номер производителя, которым будет заменен УИО текущего MAC-адреса
2	input[MAXLEN]	char	Хранение информации введенного номера

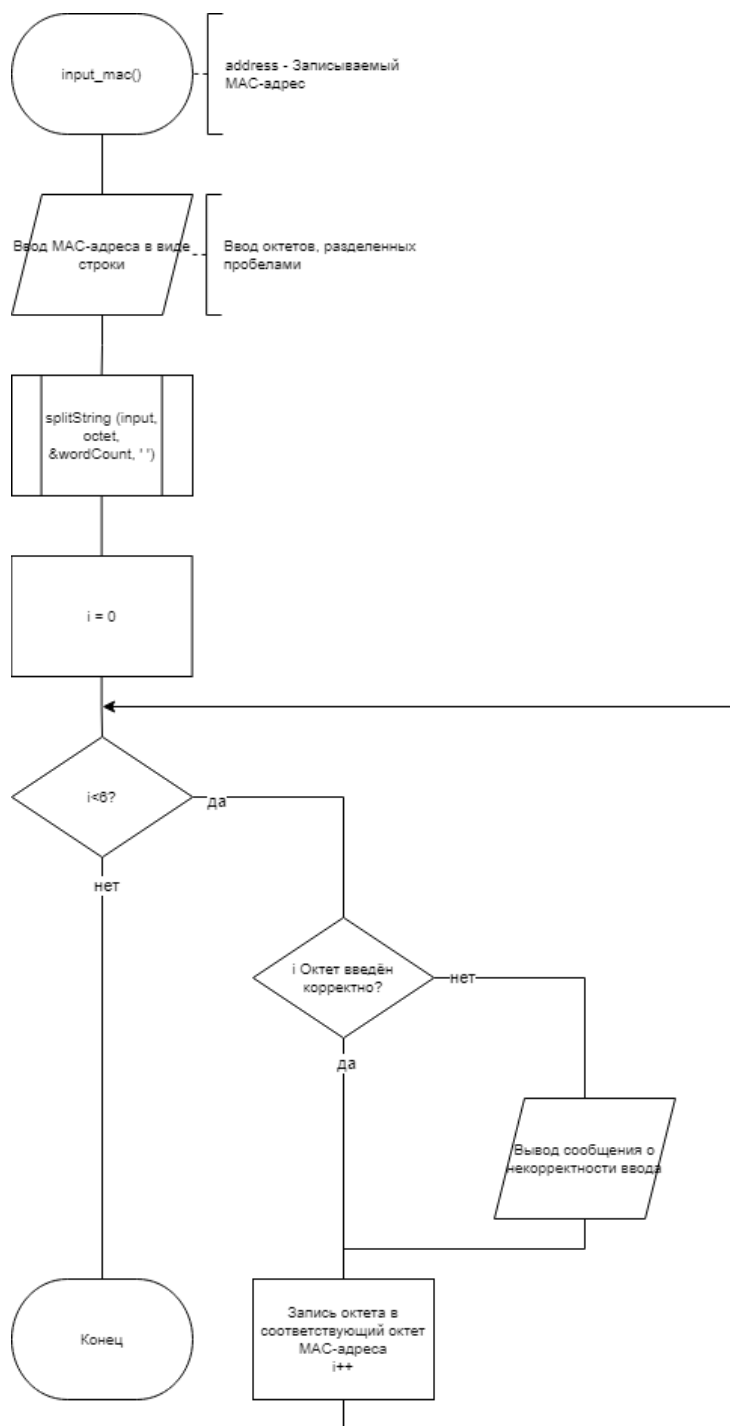
Схемы алгоритмов

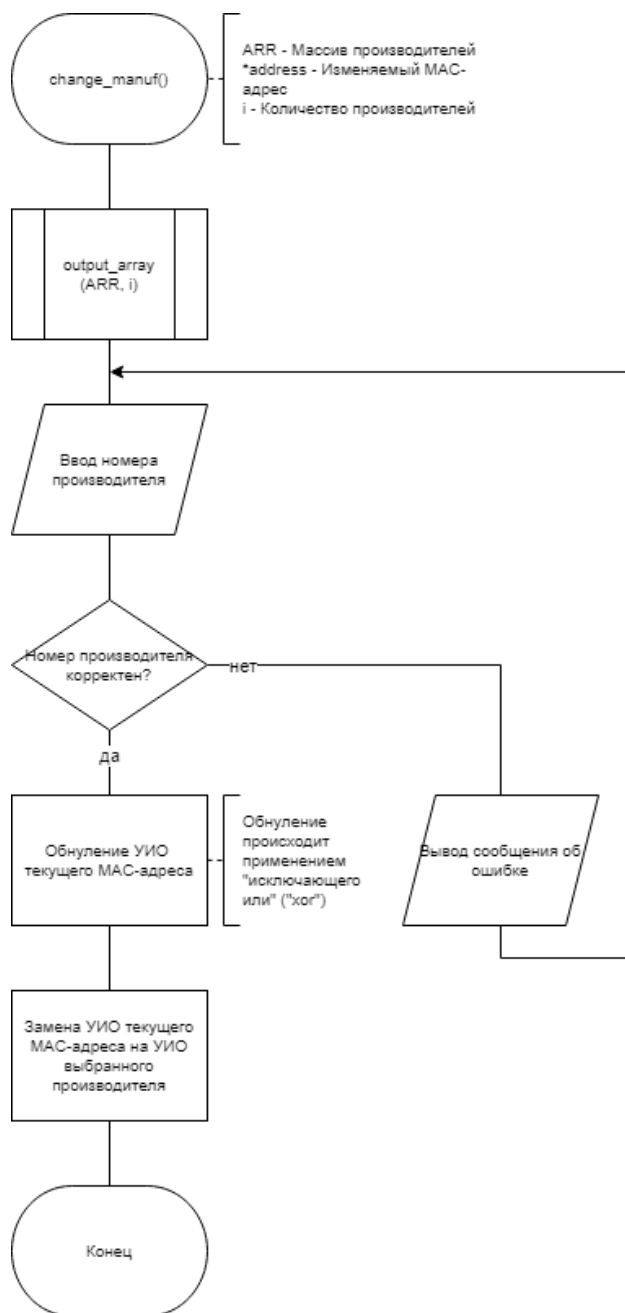












Пример 1:

Исходные данные:

Entering the MAC-Address

Enter the MAC-Address (6 Octets separated by space): 5B 8C 13 28 05 9

MAC-address: 5B:8C:13:28:05:09

Number	Manufacturer Name	Octet 1	Octet 2	Octet 3	OUI
1	ACCE Technology Corp.	EA	27	01	EA2701
2	KEVCOM LLC	3A	0E	75	3A0E75
3	GoCodeIT Inc	4A	5A	A9	4A5AA9

	4		IronWiFi		AA		14		6B		AA146B	
	5		Kandou Bus SA		BA		67		91		BA6791	
	6		OSNexus		DA		1E		56		DA1E56	
	7		Delta Computers LLC.		DA		7D		2E		DA7D2E	
	8		Global Reach Technology		6A		9D		A4		6A9DA4	

Enter the number of manufacturer that will replace your current manufacturer (1-8): 6

Результат:

Manufacturer of your MAC-address was replaced. Current MAC-address: DA:1E:56:28:05:09

Пример 2:

Исходные данные:

Entering the MAC-Address

Enter the MAC-Address (6 Octets separated by space): 13 5A 6F 0D 98 11

MAC-address: 13:5A:6F:0D:98:11

	Number		Manufacturer Name		Octet 1		Octet 2		Octet 3		OUI	
+	-----	+	-----	+	-----	+	-----	+	-----	+	-----	+
	1		ACCE Technology Corp.		EA		27		01		EA2701	
	2		KEVCOM LLC		3A		0E		75		3A0E75	
	3		GoCodeIT Inc		4A		5A		A9		4A5AA9	
	4		IronWiFi		AA		14		6B		AA146B	
	5		Kandou Bus SA		BA		67		91		BA6791	
	6		OSNexus		DA		1E		56		DA1E56	
	7		Delta Computers LLC.		DA		7D		2E		DA7D2E	
	8		Global Reach Technology		6A		9D		A4		6A9DA4	

Enter the number of manufacturer that will replace your current manufacturer (1-8): 98

Manufacturer with entered number does not exist. Please try again!

Enter the number of manufacturer that will replace your current manufacturer (1-8): 1

Результат:

Manufacturer of your MAC-address was replaced. Current MAC-address: EA:27:01:0D:98:11

Текст программы

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#define MAXLEN 128

/* !!! Need to remove pedantic mode !!! */

/* Structure -----*/

struct MACaddress{
    unsigned octet1:8;
    unsigned octet2:8;
    unsigned octet3:8;
    unsigned octet4:8;
    unsigned octet5:8;
    unsigned octet6:8;
}; /* Define structure of MAC-address */
```

```

typedef struct MACaddress MAC; /* Define structured type of MAC-address */

struct manuf; /* Define structure of manufacturer */

typedef struct manuf MNF; /* Define structured type of manufacturer */

/* Interface -----*/

void menu(); /* Output main menu */

void output_mac(MAC address); /* Output MAC-address */

void print_head(); /* Output title of sheet of manufacturers */

void output_array(MNF *ARR[], int n); /* Output array MNF type */

void clear_screen(); /* Clear the console */

/* Sort, Form and other Stuff -----*/

void input_mac(MAC *address); /* Input MAC-address */

int count_of_lines(char *filename); /* Count of lines in file */

void split_string(char *inputString, char **words, int *wordCount, char delimiter); /* Split string by
separator */

void add_to_array(MNF *ARR[], char *filename, int n, char sep); /* Adding the data of file to array */

void change_manuf(MNF *ARR[], MAC *address, int i); /* Change manufacturer of MAC-address */

/* Main program -----*/

int main(){
    int i;
    MNF **manuf = NULL;
    MAC address = {0};
    i=count_of_lines("manuf-win.txt");
    manuf=(MNF**)malloc(i*sizeof(MNF*));
    add_to_array(manuf,"manuf-win.txt",i,',');

    printf("Entering the MAC-Address\n\n");
    input_mac(&address);
    printf("\n");
    output_mac(address);
    printf("\n");
    change_manuf(manuf,&address,i);
    return 0;
}

/* Functions -----*/

struct manuf{
    char *name;
    unsigned octet1:8;
    unsigned octet2:8;
    unsigned octet3:8;
};

void menu(){
    puts("Choose the option");
    puts("0 - for EXIT program");
    puts("1 - for SHOW THE MANUFACTORER INFO");
    puts("2 - for INPUT MAC-ADDRESS");
    puts("3 - for SHOW MAC-ADDRESS");
    puts("4 - for CHANGE MFG CODE IN MAC-ADDRESS");
    printf("Enter the option: ");
}

void output_mac(MAC address){
    printf("MAC-address: %02X:%02X:%02X:%02X:%02X:%02X\n", address.octet1, address.octet2, address.octet3,
        address.octet4, address.octet5, address.octet6);
}

void print_head(){
    printf("| %6s | %25s | %8s | %8s | %8s | %6s |\n","Number","Manufacturer Name","Octet 1","Octet
        2","Octet 3","OUI");
    printf("+-----+-----+-----+-----+-----+-----+\n");
}

```

```

void output_array(MNF *ARR[], int n){
    int i;
    print_head();
    for(i=0;i<n;i++){
        printf("| %6d | %25s | %8.2X | %8.2X | %8.2X | %2.2X%2.2X%2.2X |\n",i+1, ARR[i]->name,
            ARR[i]->octet1, ARR[i]->octet2, ARR[i]->octet3, ARR[i]->octet1, ARR[i]->octet2, ARR[i]->octet3);
    }
}

void clear_screen(){
    #if defined(_WIN32) || defined(_WIN64)
        system("cls");
    #else
        system("clear");
    #endif
}

void input_mac(MAC *address){
    char input[MAXLEN], *octet[6];
    unsigned int a;
    int i, wordCount;
    printf("Enter the MAC-Address (6 Octets separated by space): ");
    fgets(input, MAXLEN, stdin); /* Read the string */
    input[strcspn(input, "\n")] = 0; /* Delete symbol of 'enter' */
    split_string(input,octet,&wordCount,' ');
    for (i=0; i<6; i++) {
        if (sscanf(octet[i], "%x", &a) != 1 || a > 255 || strlen(octet[i])>2) printf("Invalid Input. Octet%d
            will be written incorrectly in the MAC address.\n", i+1);

        switch (i) {
            case 0: address->octet1 = (unsigned char)a; break;
            case 1: address->octet2 = (unsigned char)a; break;
            case 2: address->octet3 = (unsigned char)a; break;
            case 3: address->octet4 = (unsigned char)a; break;
            case 4: address->octet5 = (unsigned char)a; break;
            case 5: address->octet6 = (unsigned char)a; break;
        }
    }
}

int count_of_lines(char *filename) {
    char line[MAXLEN];
    int n=0;

    FILE *file = fopen(filename, "r");

    if (file!=NULL) {
        while((fgets(line,MAXLEN,file))!=NULL) n++;
        rewind(file);
    } else {
        printf("Error opening the file.\n");
    }
    return n;
}

void split_string(char *inputString, char **words, int *wordCount, char delimiter) {
    int wordIndex = 0, wordStart=0, wordLength=0, inWord=0, i, len;

    for (i = 0; i <= strlen(inputString); i++) {
        if ((inputString[i] == delimiter || inputString[i] == '\0') && (inWord==1)) {
            words[wordIndex] = (char *)malloc(wordLength + 1);
            strncpy(words[wordIndex], inputString + wordStart, wordLength);
            words[wordIndex][wordLength] = '\0';
            wordIndex++;
            inWord = 0;
        } else {
            if (inWord==0) {
                wordStart = i;
                wordLength = 1;
                inWord = 1;
            } else {
                wordLength++;
            }
        }
    }
    *wordCount = wordIndex;
    /* For correct work on MAC and ?LINUX? */
    for (i = 0; i < *wordCount; i++) {
        len = strlen(words[i]);

```

```

        if (words[i][len - 1] == '\r') words[i][len - 1] = '\0';
    }
}

void add_to_array(MNF *ARR[], char *filename, int n, char sep) {
    char line[MAXLEN];
    char **TMP = NULL;
    int wordCount = 0, i, j;
    unsigned int temp_octet1, temp_octet2, temp_octet3;

    FILE *file = fopen(filename, "r");

    if (file != NULL) {
        for (i = 0; i < n; i++) {
            fgets(line, MAXLEN, file);
            TMP = (char **)malloc(MAXLEN * sizeof(char *));
            split_string(line, TMP, &wordCount, sep);
            if (TMP != NULL) {
                ARR[i] = (MNF*)malloc(sizeof(MNF));

                ARR[i]->name = (char *)malloc(strlen(TMP[0]) + 1);
                strcpy(ARR[i]->name, TMP[0]);

                sscanf(TMP[1], "%x", &temp_octet1);
                sscanf(TMP[2], "%x", &temp_octet2);
                sscanf(TMP[3], "%x", &temp_octet3);

                ARR[i]->octet1 = temp_octet1;
                ARR[i]->octet2 = temp_octet2;
                ARR[i]->octet3 = temp_octet3;
            }
            for (j = 0; j < wordCount; j++) {
                free(TMP[j]);
            }
            free(TMP);
        }
    } else {
        printf("Error opening the file.\n");
    }
    fclose(file);
}

void change_manuf(MNF *ARR[], MAC *address, int i){
    char input[MAXLEN];
    int number;
    output_array(ARR, i);
    do{
        printf("\nEnter the number of manufacturer that will replace your current manufacturer (1-%d): ", i);
        fgets(input, MAXLEN, stdin); /* Read the string */
        input[strcspn(input, "\n")] = 0; /* Delete symbol of 'enter' */
        if (sscanf(input, "%i", &number) == 1 && number < i){
            address->octet1 ^= address->octet1;
            address->octet2 ^= address->octet2;
            address->octet3 ^= address->octet3;
            printf("\nManufacturer of your MAC-address was replaced. Current ");
            address->octet1 = ARR[number-1]->octet1;
            address->octet2 = ARR[number-1]->octet2;
            address->octet3 = ARR[number-1]->octet3;
            output_mac(*address);
        } else printf("\nManufacturer with entered number does not exist. Please try again!\n");
    } while (sscanf(input, "%i", &number) != 1 || number > i);
}

```

Примеры выполнения программы

Исходный текстовый файл и данные:

ACCE Technology Corp.;EA;27;01	Number	Manufacturer Name	Octet 1	Octet 2	Octet 3	OUI
KEVCOM LLC;3A;0E;75	1	ACCE Technology Corp.	EA	27	01	EA2701
GoCodeIT Inc;4A;5A;A9	2	KEVCOM LLC	3A	0E	75	3A0E75
IronWiFi;AA;14;6B	3	GoCodeIT Inc	4A	5A	A9	4A5AA9
Kandou Bus SA;BA;67;91	4	IronWiFi	AA	14	6B	AA146B
OSNexus;DA;1E;56	5	Kandou Bus SA	BA	67	91	BA6791
Delta Computers LLC.;DA;7D;2E	6	OSNexus	DA	1E	56	DA1E56
Global Reach Technology;6A;9D;A4	7	Delta Computers LLC.	DA	7D	2E	DA7D2E
	8	Global Reach Technology	6A	9D	A4	6A9DA4

Пример 1:

```
Entering the MAC-Address
Enter the MAC-Address (6 Octets separated by space): 5B 8C 13 28 05 9
MAC-address: 5B:8C:13:28:05:09

| Number | Manufacturer Name | Octet 1 | Octet 2 | Octet 3 | OUI |
+-----+-----+-----+-----+-----+-----+
| 1 | ACCE Technology Corp. | EA | 27 | 01 | EA2701 |
| 2 | KEVCOM LLC | 3A | 0E | 75 | 3A0E75 |
| 3 | GoCodeIT Inc | 4A | 5A | A9 | 4A5AA9 |
| 4 | IronWiFi | AA | 14 | 6B | AA146B |
| 5 | Kandou Bus SA | BA | 67 | 91 | BA6791 |
| 6 | OSNexus | DA | 1E | 56 | DA1E56 |
| 7 | Delta Computers LLC. | DA | 7D | 2E | DA7D2E |
| 8 | Global Reach Technology | 6A | 9D | A4 | 6A9DA4 |

Enter the number of manufacturer that will replace your current manufacturer (1-8): 6
Manufacturer of your MAC-address was replaced. Current MAC-address: DA:1E:56:28:05:09
Process returned 0 (0x0) execution time : 3.931 s
Press any key to continue.
```

Пример 2:

```
Entering the MAC-Address
Enter the MAC-Address (6 Octets separated by space): 13 5A 6F 0D 98 11
MAC-address: 13:5A:6F:0D:98:11

| Number | Manufacturer Name | Octet 1 | Octet 2 | Octet 3 | OUI |
+-----+-----+-----+-----+-----+-----+
| 1 | ACCE Technology Corp. | EA | 27 | 01 | EA2701 |
| 2 | KEVCOM LLC | 3A | 0E | 75 | 3A0E75 |
| 3 | GoCodeIT Inc | 4A | 5A | A9 | 4A5AA9 |
| 4 | IronWiFi | AA | 14 | 6B | AA146B |
| 5 | Kandou Bus SA | BA | 67 | 91 | BA6791 |
| 6 | OSNexus | DA | 1E | 56 | DA1E56 |
| 7 | Delta Computers LLC. | DA | 7D | 2E | DA7D2E |
| 8 | Global Reach Technology | 6A | 9D | A4 | 6A9DA4 |

Enter the number of manufacturer that will replace your current manufacturer (1-8): 98
Manufacturer with entered number does not exist. Please try again!
Enter the number of manufacturer that will replace your current manufacturer (1-8): 1
Manufacturer of your MAC-address was replaced. Current MAC-address: EA:27:01:0D:98:11
Process returned 0 (0x0) execution time : 43.745 s
Press any key to continue.
```

Вывод

В результате выполнения работы изучена и освоена работа с битовыми полями в структурах.