

Министерство образования Республики Беларусь

Учреждение образования  
«Белорусский государственный университет информатики и радиоэлектроники»

Кафедра электронных вычислительных машин

Отчёт  
по лабораторной работе № 8  
«Рекурсия. Указатели на функции»

по курсу:  
«Основы алгоритмизации и программирования»

Выполнил:  
Студент группы 950503  
Полховский А.Ф.

Проверил:  
Дубовский А.Л.

Минск 2019

Цель работы: Научиться использовать в программах указатели на функции

Вариант 4

Условие:

11) Дан массив строк. Найти строку, содержащую слово с наибольшим числом символов. Память под массив строк отводить динамически. Использовать указатели на функции.

Код программы на языке Си:

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

int n;

int getIndexOfStringWithMaxWord(char **strings) {
    int indexOfStringWithMaxWord = 0;
    int maxLength = 0;
    for (int i = 0; i < n; i++) {
        int j = 0;
        int maxWordLengthInString = 0;
        int wordLength = 0;
        while (strings[i][j] != '\0') {
            if (strings[i][j] != ' ' && strings[i][j] != '.' && strings[i][j] !=
',') {
                wordLength++;
            }
            if (strings[i][j] == ' ' || strings[i][j] == '.' || strings[i][j] ==
', ' || strings[i][j + 1] == '\0') { // i+1 для while(не сработает посл раз)
                if (wordLength > maxWordLengthInString) {
                    maxWordLengthInString = wordLength;
                }
                wordLength = 0;
            }
            j++;
        }
        if (maxWordLengthInString > maxLength) {
            maxLength = maxWordLengthInString;
            indexOfStringWithMaxWord = i;
        }
    }

    return indexOfStringWithMaxWord;
}

int main() {
    setlocale(LC_ALL, "RUSSIAN");
    printf("Input n (number of strings)\n");
    scanf_s("%d", &n);

    int (*pointer)(char**) = getIndexOfStringWithMaxWord; // указатель на
функцию

    char** strings = (char**)malloc((n) * sizeof(char*)); // выделение памяти для
строк
    for (int j = 0; j < n; j++) {
        int i = 0;
        int size = 5;
        printf("Input %d string\n", j);
```

```

        strings[j] = (char*)calloc(size, sizeof(char)); //выделение памяти для
каждой строки
        rewind(stdin);
        do {
            strings[j][i] = getchar();
            if (strings[j][i] == '\n') break;
            i++;
            if (i >= size) {
                size *= 2;
                strings[j] = (char*)realloc(strings[j], size);
            }
        } while (strings[j][i - 1] != 32 || strings[j][i - 1] != EOF ||
strings[j][i - 1] != '\r');
        strings[j][i] = '\0';
    }

    for (int i = 0; i < n; i++) {
        printf("\n%d string = %s", i, strings[i]);
    }

    int indexresultString;

    indexresultString = (*pointer)(strings); //указатель на функцию
    printf("\nResult string = %s", strings [indexresultString]);

    getchar();
    return 0;
}

```

Вывод: ознакомился с основными теоритическими сведениями по данной лабораторной работе; а также разработал, написал и отладил программу 4-ого варианта.