**Тарасов, УТС-11, Вариант-7.**

**Задание:**

В динамическом текстовом массиве определить местоположение «точек». Для этого создать функцию **find\_point(),** возвращающую массив результатов (массив указателей). В функции **main(),** имея массив указателей, после каждой точки добавить символ «enter». Распечатать измененный массив.

Привести два решения, разместить указатель на массив результатов:

а) в функции **main();** б) в функции **find\_point()**

**Программная реализация:**

**Пункт a):**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <stdio.h>

using namespace std;

//Считаем количество точек:

int numOfPoints(char\* str)

{

int count = 0;

for (int i = 0; i < strlen(str); ++i)

{

if (str[i] == '.')

++count;

}

return count;

}

//Функция указателей:

char\*\* find\_point(char\* str, char\*\* points)

{

int j = 0;

for (int i = 0; i < strlen(str); ++i)

{

if (str[i] == '.') \* (points + (j++)) = &str[i];

}

return points;

}

void main()

{

setlocale(LC\_ALL, "Rus");

cout << "Введите текст:" << endl;

char buffer[255];

gets\_s(buffer);

char\* str = new char[255];

strcpy(str, buffer);

str[strlen(buffer)] = '\0';

char\*\* points = new char\* [numOfPoints(str)];

points = find\_point(str, points);

for (int i = 0; i < numOfPoints(str); ++i)

{

int j = 0;

char bff[255];

while (\*(\*(points + i) + j + 1) != '\0')

{

bff[j] = (\*(\*(points + i) + j + 1));

++j;

}

bff[j] = '\0';

\*(\*(points + i) + 1) = '\n';

for (j = 1; i + j < numOfPoints(str); ++j)++ \* (points + i + j);

j = 0;

while (bff[j] != '\0')

{

(\*(\*(points + i) + j + 2)) = bff[j];

++j;

}

(\*(\*(points + i) + j + 2)) = '\0';

}

cout << endl << "Вывод:" << endl;

puts(str);

}

**Пункт б):**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <stdio.h>

using namespace std;

//Считаем количество точек:

int numOfPoints(char\* str)

{

int count = 0;

for (int i = 0; i < strlen(str); ++i)

{

if (str[i] == '.')

++count;

}

return count;

}

//Функция указателей:

char\*\* find\_point(char\* str)

{

char\*\* points = new char\* [numOfPoints(str)];

int j = 0;

for (int i = 0; i < strlen(str); ++i)

{

if (str[i] == '.') \* (points + (j++)) = &str[i];

}

return points;

}

void main()

{

setlocale(LC\_ALL, "Rus");

cout << "Введите текст:" << endl;

//Вводим текст:

char buffer[255];

gets\_s(buffer);

cout << endl;

//Записываем в массив:

char\* str = new char[255];

strcpy(str, buffer);

str[strlen(buffer)] = '\0';

//Создаем массив, куда будем все выгружать (двумерный):

char\*\* points = find\_point(str);

//Переносим строки:

for (int i = 0; i < numOfPoints(str); ++i)

{

int j = 0;

char bff[255];

while (\*(\*(points + i) + j + 1) != '\0')

{

bff[j] = (\*(\*(points + i) + j + 1));

++j;

}

bff[j] = '\0'; //избавляемся от мусора

\*(\*(points + i) + 1) = '\n';

for (j = 1; i + j < numOfPoints(str); ++j)++ \* (points + i + j);

j = 0;

while (bff[j] != '\0')

{

(\*(\*(points + i) + j + 2)) = bff[j];

++j;

}

(\*(\*(points + i) + j + 2)) = '\0';

}

cout << endl << "Вывод:" << endl;

puts(str);

}