**БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Международный Институт Дистанционного Образования

Кафедра “Информационные системы и технологии”

Лабораторная работа номер 3 по дисциплине «Системное программирование»

Тема работы:

**«Программирование на Си»**

Выполнил:   
студент 3 курса, гр. 41703120  
Реут Владислав Леонидович

Проверил: Бумай А.Ю.

Минск 2022

**ВВЕДЕНИЕ**

Язык программирования С (си) является одним из самых популярных и распространенных языков. Он представляет компилируемый язык программирования общего назначения со статической типизацией, разработанный в 1969—1973 годах в компании Bell Labs программистом Деннисом Ритчи (Dennis Ritchie).

Язык С нередко называют языком программирования "среднего уровня" или даже "низкого уровня", так как он сочетает элементы языков высокого уровня с функциональностью и производительностью ассемблера и работает близко к аппаратной части компьютера. В итоге мы можем манипулировать данными на низком уровне и при этом использовать высокоуровневые конструкции для управления работы программы.

Первоначально язык С предназначался для написания операционной системы Unix. Впоследствии Си стал одним из популярных языков, а его основной сферой применения стало системное программирование, в частности, создание операционных систем, драйверов, различных утилит, антивирусов и т.д. К слову сказать, Linux большей частью написан на Си. Однако только системным программированием применение данного языка не ограничивается. Данный язык можно использовать в программах любого уровня, где важны скорость работы и производительность. Так, мы можем писать с помощью Си и прикладные приложения, и даже веб-сайты (используя технологию CGI - Common Gateway Interface). Но, конечно, для создания графического интерфейса и веб-приложений, как правило, выбираются более подходящие инструменты и технологии, но тем не менее круг использования Си довольно широк. Это в немалой степени определило популярность языка. Например, в известном рейтинге языков программирования TIOBE язык С долгое время уверенно удерживает второе место.

Несмотря на большие возможности язык Си одновременно довольно прост. Он не содержит много конструкций, библиотек, его легко осваивать и изучать. Поэтому нередко его выбирают в качестве языка для изучения в целом программированию.

Си является компилируемым языком, а это значит, что компилятор транслирует исходный код на Си в исполняемый файл, который содержит набор машинных инструкций. Но разные платформы имеют свои особенности, поэтому скомпилированные программы нельзя просто перенести с одной платформы на другую и там уже запустить. Однако на уровне исходного кода программы на Си обладают переносимостью, а наличие компиляторов, библиотек и инструментов разработки почти под все распространенные платформы позволяет компилировать один и тот же исходный код на Си в приложения под эти платформы.

Развитие Си оказало большое влияние в целом на развитие языков программирования. В частности, его синтаксис стал основой для таких языков как С++, С#, Java, PHP, JavaScript. Особо следует сказать про связь с C++. C++ напрямую произошёл от Си. Но впоследствии их развитие происходило отдельно друг от друга, и даже появилась несовместимость между ними. Стандарт C99 добавил в язык Си ряд конфликтующих с C++ особенностей. В итоге в настоящее время оба языка являются фактически самодостаточными и развиваются независимо.

**Вариант №5**

Массив размерностью MxN. Необходимо дополнить его (M+1)-й строкой и (N+1)-м столбцом, в которых записать суммы элементов соответствующих строк и столбцов. В элементе aM+1,N+1 должна храниться сумма всех элементов массива.

#include <stdio.h>

#include <stdlib.h>

#define M 5

#define N 5

// Функция выделения динамической памяти для двумерного массива

int\*\* malloc2DArray( int m, int n)

{

int\*\* arr = (int\*\*) malloc(m\*sizeof(int\*));

for (int i = 0; i < m; ++i)

arr[i] = (int\*) malloc(n\*sizeof(int));

return arr;

}

// Функция освобождения памяти

void free2DArray(int\*\* arr, const int m)

{

for (int i = 0; i < m; ++i)

free(arr[i]);

free(arr);

}

// Печать двумерного массива

void print2DArray(int\*\* arr, int m , int n)

{

for (int i = 0; i < m; ++i)

{

for (int j = 0; j < n; ++j)

printf("%d ", arr[i][j]);

printf("\n");

}

}

// Функция суммирования строк

int sumRow(int \*\*arr, int row, int cols)

{

int sum = 0;

for (int j = 0; j < cols; ++j)

sum += arr[row][j];

return sum;

}

// Функция суммирования столбцов

int sumCol(int \*\*arr, int col, int rows)

{

int sum = 0;

for (int i = 0; i < rows; ++i)

sum += arr[i][col];

return sum;

}

int main()

{

// создаём новую матрицу MxN

int\*\* mtrx = malloc2DArray(M+1, N+1);

// заполняем

for (int i = 0; i < M; ++i)

for (int j = 0; j < N; ++j)

mtrx[i][j] = j;

// выводим

printf("Matrix\n");

print2DArray(mtrx, M, N);

int rowsSum = 0;

for (int i = 0; i < M; ++i)

{

mtrx[i][N] = sumRow(mtrx, i, N);

rowsSum += mtrx[i][N];

}

int colsSum = 0;

for (int j = 0; j < N; ++j)

{

mtrx[M][j] = sumCol(mtrx, j, M);

colsSum += mtrx[M][j];

}

mtrx[M][N] = rowsSum + colsSum;

// выводим

printf("Matrix\n");

print2DArray(mtrx, M+1, N+1);

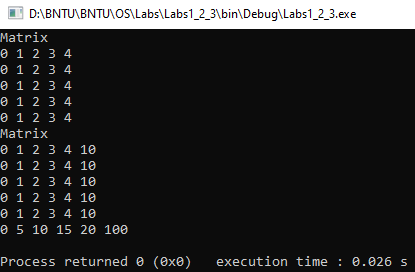
// освобождаем выделенную память

free2DArray(mtrx, M);

return 0;

}

Результат работы функции:



**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. ЛАБОРАТОРНЫЙ ПРАКТИКУМ ПО КУРСУ СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ
2. Сайт metanit.com