|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Ing. Claudia Rodríguez Espino |
| *Asignatura:* | Fundamentos de Programación |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | 11 |
| *Integrante(s):* | Alejandro Meneses Mercado |
|  |  |
|  |  |
| *Semestre:* | 2018-1 |
| *Fecha de entrega:* | 3 de noviembre del 2017 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Práctica 11: Arreglos unidimensionales y multidimensionales

Objetivo:

Reconocer la importancia y utilidad de los arreglos, en la elaboración de programas que resuelvan problemas que requieran agrupar datos del mismo tipo, así como trabajar con arreglos tanto unidimensionales como multidimensionales.

Desarrollo:

Empezamos retomando el concepto ya visto previamente de arreglo el cual es un conjunto de datos del mismo tipo con un tamaño fijo definido al momento de crearse.

Vimos que los arreglos pueden ser unidimensionales o multidimensionales.

Para empezar revisamos lo que eran los arreglos unidimensionales los cuales estan conformados de n elementos, la primera localidad del arreglo corresponde al índice 0 y desciende hasta la última que corresponde al índice n-1.

También vimos que su sintaxis es la sig:

* tipoDeDato nombre[tamaño]

Y que los arreglos pueden ser de tipo entero, real, carácter o estructura.

A continuación vimos lo que era un apuntador el cual es una variable que contiene la dirección de una variable, es decir, hace referencia a la localidad de memoria de otra variable.

Para declarar una variable apuntador se debe iniciar con el carácter \*.

Por lo cual su sintaxis es:

* TipoDeDato \*apuntador, variable;
* apuntador = &variable;

Después vimos que cuando hay un ampersand antes de una variable, se accede a la dirección de memoria de la misma.

Aparte vimos que los apuntadores solo pueden apuntar a direcciones de memoria del mismo tipo de dato con el que fueron declarados y para que podamos acceder al contenido de dicha dirección, a la variable apuntador se le antepone \*.

Y por último revisamos lo que son los arreglos multidimensionales, que son arreglos con varias dimensiones.

También revisamos su sintaxis la cual es:

* tipoDato nombre[ tamaño ][ tamaño ]…[tamaño];

Vimos que se puede considerar que la primera dimensión corresponde a los renglones, la segunda a las columnas, la tercera al plano, y así sucesivamente.

Y para terminar vimos que se puede recorrer un arreglo multidimensional con apuntadores.

Como actividad tuvimos que elaborar 3 programas usando lo visto en la práctica.

1. Programa que suma dos matrices de 2x2

Código:

#include<stdio.h>

int i,j,A[3][3],B[3][3],C[3][3];

main()

{

printf("\t\nEste programa Suma dos matrices de 2x2\n");

printf("\n\tIngresa las entradas de la Matriz A\n\n");

for(i=1;i<=2;i++)

{

for(j=1;j<=2;j++)

{

printf("Da un valor para A[%d][%d]: ",i,j);

scanf("%i",&A[i][j]);

}

}

printf("\n\tIngresa las entradas de la Matriz B\n\n");

for(i=1;i<=2;i++)

{

for(j=1;j<=2;j++)

{

printf("Da un valor para B[%d][%d]: ",i,j);

scanf("%i",&B[i][j]);

}

}

printf("\n\t\tEl resultado de la suma de ambas Matices es:\n\n");

for(i=1;i<=2;i++)

{

for(j=1;j<=2;j++)

{

C[i][j]=A[i][j]+B[i][j];

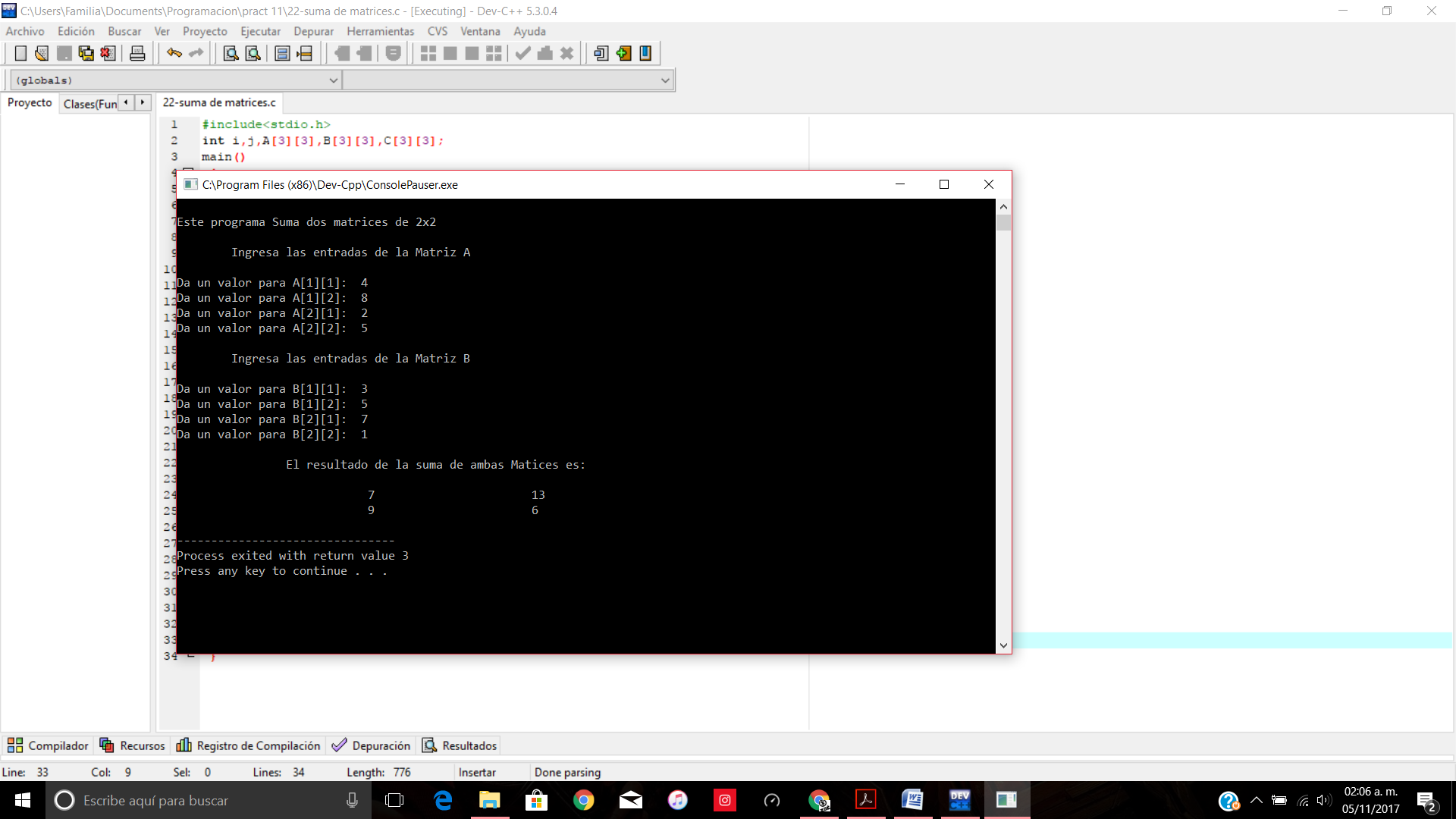
printf("\t\t\t %d",C[i][j]);

}

printf("\t\t\t\n");

}

}



1. Programa que multiplica una matriz 2x2 por una escalar

Código:

#include<stdio.h>

int i,j,A[3][3],B[3][3],c;

main()

{

printf("\t\nEste programa multiplica una matriz de 2x2 por una escalar\n");

printf("\n\tIngresa las entradas de la Matriz A\n\n");

for(i=1;i<=2;i++)

{

for(j=1;j<=2;j++)

{

printf("Da un valor para A[%d][%d]: ",i,j);

scanf("%i",&A[i][j]);

}

}

printf("\nIntroduce el valor de la escalar: ");

scanf("%d",&c);

printf("\n\t\tEl resultado de la multiplicacion de la matriz por escalar es:\n\n");

for(i=1;i<=2;i++)

{

for(j=1;j<=2;j++)

{

B[i][j]=A[i][j]\*c;

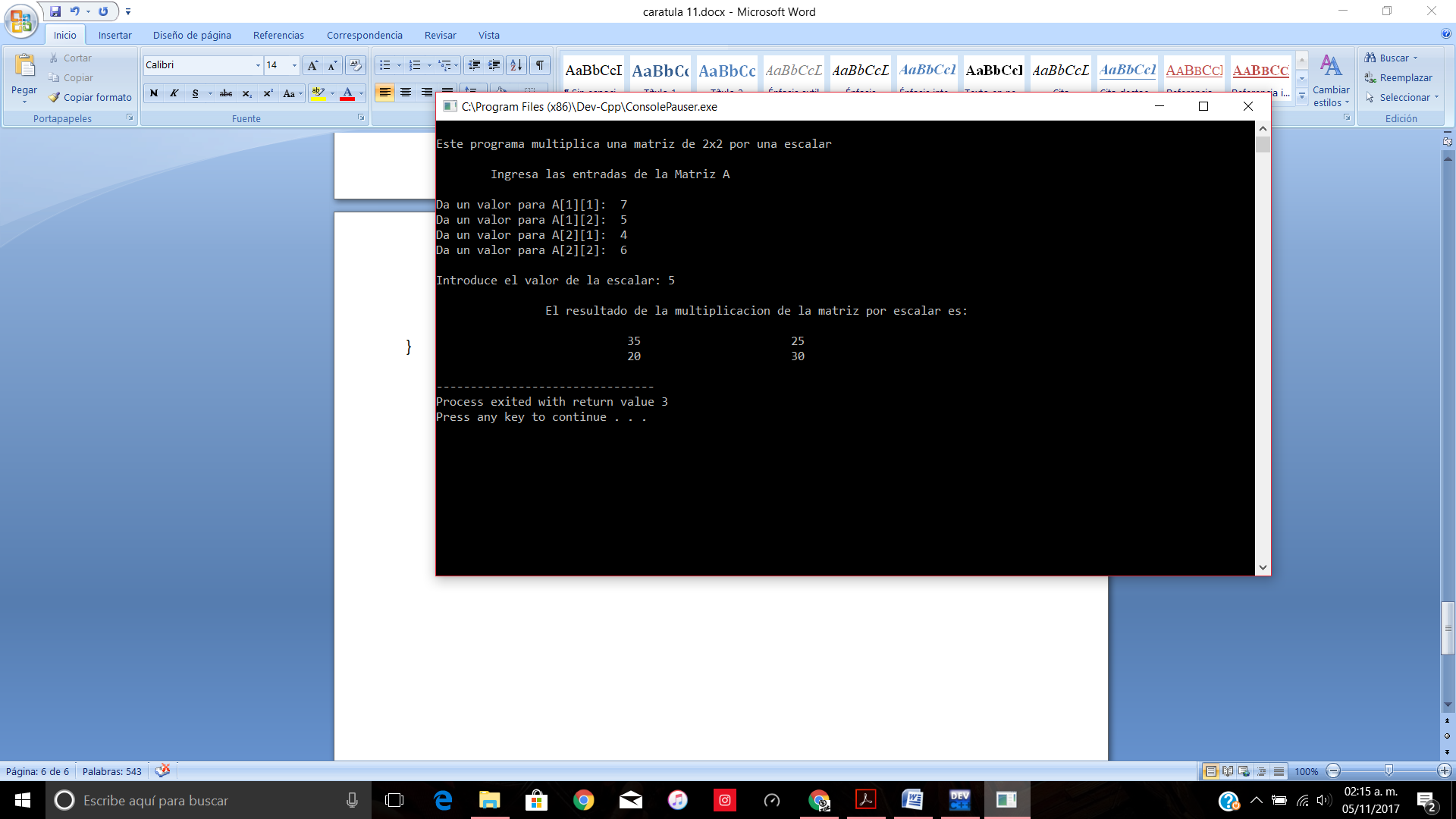
printf("\t\t\t %d",B[i][j]);

}

printf("\t\t\t\n");

}

}



1. Programa que muestra los meses con apuntadores para sacar el promedio de los gastos del año

Código:

#include <stdio.h>

char \*meses[] = {"Enero","Febrero","Marzo","Abril","Mayo","Junio","Julio","Agosto",

"Septiembre","Octubre","Noviembre","Diciembre"};

float mes[15],d=0,p=0;

int a;

main()

{

for(a=0;a<12;a++)

{

printf("Escribe los gastos de %s:\t",meses[a]);

scanf("%f",&mes[a]);

}

printf("\n\nLos gastos de %s son: %.2f",meses[5], mes[5]);

printf("\n\nLos gastos de %s son: %.2f\n",meses[11], mes[11]);

for(a=0;a<12;a++)

{

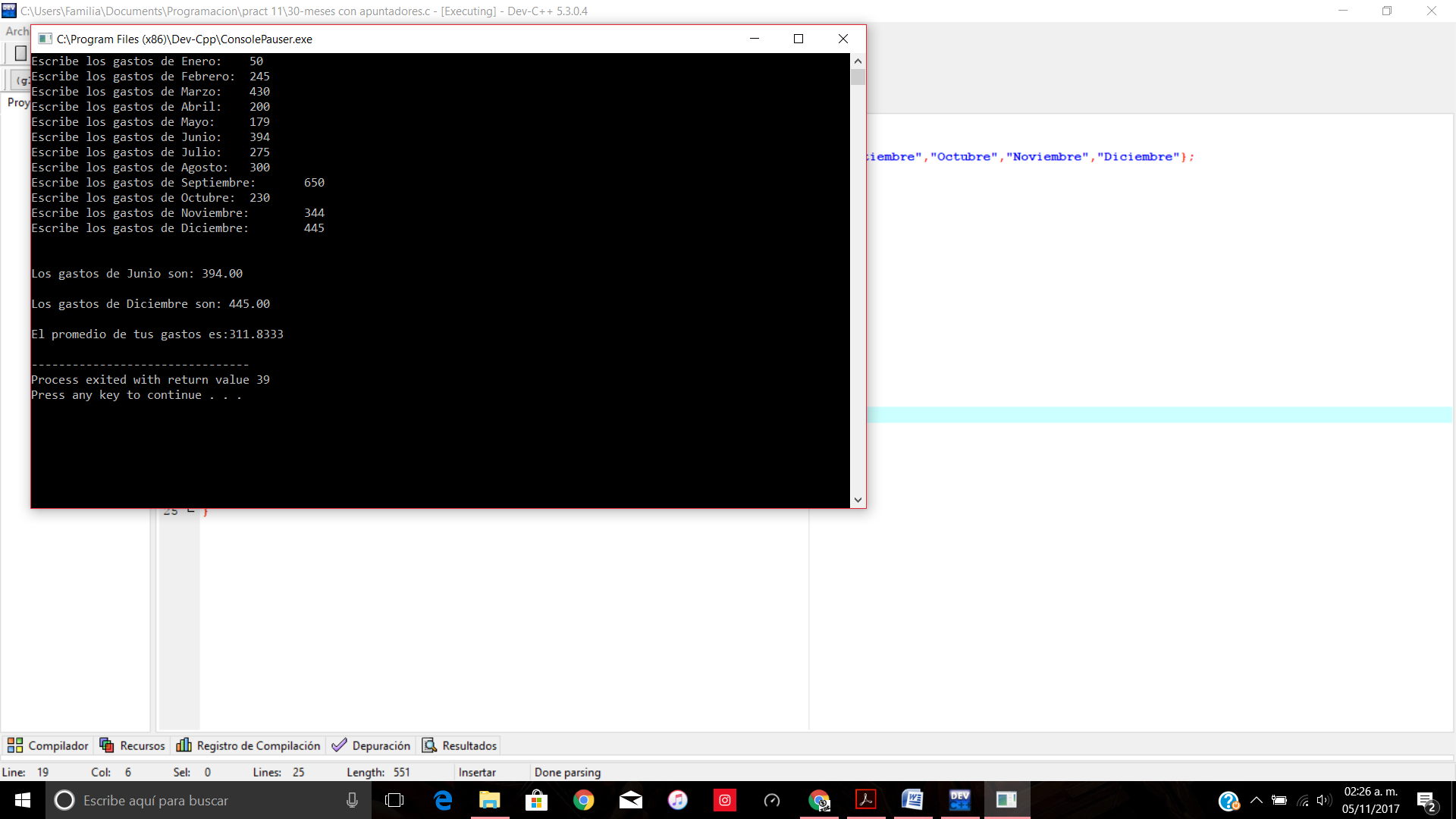
d=(d+mes[a]);

}

p=d/12;

printf("\nEl promedio de tus gastos es:%.4f\n",p);

}



Conclusiones:

Gracias a esta práctica pudimos comprender mejor la función de un arreglo y entender un poco mas el funcionamiento de un apuntador que nos servira en futuros programas que realicemos.