

	Carátula para entrega de prácticas	
Facultad de Ingeniería		Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: M.C. ALEJANDRO ESTEBAN PIMENTEL ALARCON

Asignatura: FUNDAMENTOS DE PROGRAMACION

Grupo: 3

No de Práctica(s): PRACTICA 9

Integrante(s): 1

*No. de Equipo de
cómputo empleado:* Equipo 3

*No. de Lista o
Práctica:* 420054913

Semestre: 2020-1

Fecha de entrega: 14 DE OCTUBRE 2019

Observaciones: Muy bien. Pero cuida más el formato de tu caratula.
Apegate al formato original

CALIFICACIÓN:

10

OBJETIVO

ELABORAR PROGRAMAS EN C PARA LA RESOLUCIÓN DE PROBLEMAS BÁSICOS QUE INCLUYAN LAS ESTRUCTURAS DE REPETICIÓN Y LA DIRECTIVA DEFINE.

En esta práctica se probaran los diferentes tipos de iteraciones que tiene el lenguaje de programación y como estas hacen sus procesos de formas diferentes pero realizando la misma acción.

Para cada uno de los siguientes problemas, elegir un tipo de ciclo y resolverlo. Al final, debe usar los tres tipos de ciclos y usar define por lo menos una vez.

Hacer un programa que pida un número y muestre su tabla de multiplicar (hasta el 10).

While

Se hace un ciclo con while que solo se detiene hasta llegar a 10

F:\practicas\practica 9\multiplicar.c - Notepad++

Archivo Editar Buscar Vista Codificación Lenguaje Configuración Herramientas Macro
Ejecutar Plugins Ventana ?

main.c main1.c main2.c primo.c **multiplicar.c** sumaydiv.c primodo.c

```
1  #include <stdio.h>
2  int main() {
3      int a, re;
4      int i = 0;
5      # define b 10
6      scanf("%i", &a);
7      while(i<=b){
8          re = a * i;
9          printf ("%i * %i = %i \n", a, i, re);
10         i++;
11     }
12     return 0;
13 }
```

length: 18 Ln: 13 Col: 2 Sel: 0 | 0 Windows (CR LF) UTF-8 INS

```
~ /del9

amara@DESKTOP-3AT0670 ~
$ cd 9del
-bash: cd: 9del: No such file or directory

amara@DESKTOP-3AT0670 ~
$ cd del9

amara@DESKTOP-3AT0670 ~/del9
$ gcc multiplicar.c -o m

amara@DESKTOP-3AT0670 ~/del9
$ ./m 5
5
5 * 0 = 0
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50

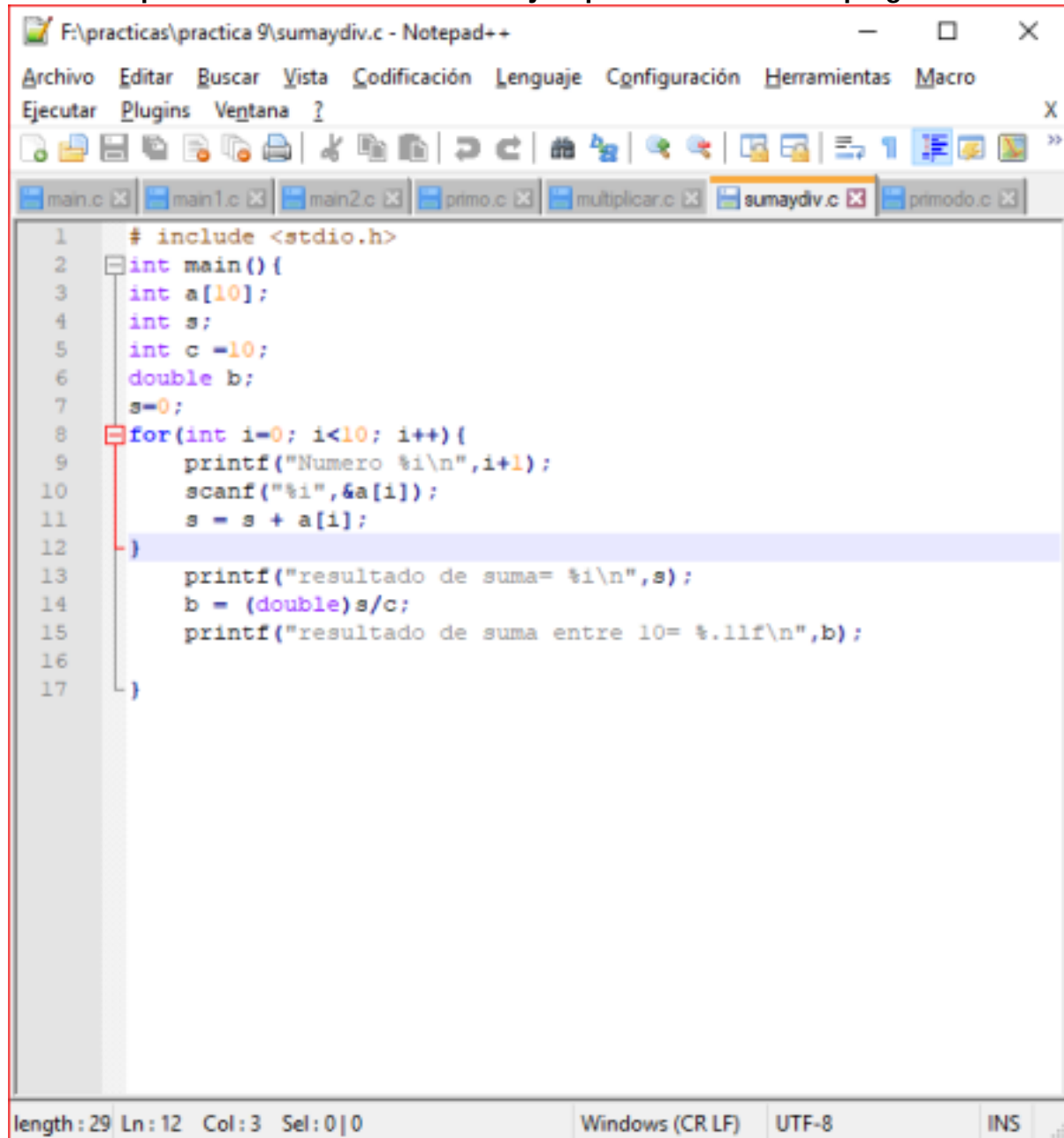
amara@DESKTOP-3AT0670 ~/del9
$ ./m
9
9 * 0 = 0
9 * 1 = 9
9 * 2 = 18
9 * 3 = 27
9 * 4 = 36
9 * 5 = 45
9 * 6 = 54
9 * 7 = 63
9 * 8 = 72
9 * 9 = 81
9 * 10 = 90

amara@DESKTOP-3AT0670 ~/del9
$ |
```

Hacer un programa que pida y lea 10 números y muestre su suma y su promedio.

For

El for hace un recorrido en un array que llena y va a la vez sumando con los valores anteriores para mostrar la suma de todos y su promedio al final del programa.



```
1  #include <stdio.h>
2  int main() {
3      int a[10];
4      int s;
5      int c = 10;
6      double b;
7      s = 0;
8      for(int i=0; i<10; i++){
9          printf("Numero %i\n", i+1);
10         scanf("%i", &a[i]);
11         s = s + a[i];
12     }
13     printf("resultado de suma= %i\n", s);
14     b = (double)s/c;
15     printf("resultado de suma entre 10= %.11f\n", b);
16
17 }
```

length: 29 Ln: 12 Col: 3 Sel: 0|0 Windows (CR LF) UTF-8 INS

```
amara@DESKTOP-3AT0670 ~/de19
$ gcc sumaydiv.c -o sm

amara@DESKTOP-3AT0670 ~/de19
$ ./sm
1
Numero 1
Numero 2
2
Numero 3
3
Numero 4
5
Numero 5
80
Numero 6
10
Numero 7
11
Numero 8
12
Numero 9
13
Numero 10
56
resultado de suma= 193
resultado de suma entre 10= 19.3

amara@DESKTOP-3AT0670 ~/de19
$
```

Hacer un programa que pida un número e indique si es primo o no.

Do While

Lo que hace el do while aqui es checar con cuantos números el valor ingresado se puede dividir dejando residuo 0, y según esto decir si es un numero primo o no.

```
F:\practicas\practica 9\primodo.c - Notepad++
Archivo  Editar  Buscar  Vista  Codificación  Lenguaje  Configuración  Herramientas  Macro
Ejecutar  Plugins  Ventana  ?

main.c  main1.c  main2.c  primodo.c  multiplicar.c  sumaydiv.c  primodo.c

1  #include <stdio.h>
2  int main() {
3      int p;
4      scanf("%i", &p);
5      int i = 1;
6      int c = 0;
7
8      if(p == 1 || p == 0 || p < 0) {
9          printf("no es numero primo");
10         else{
11             do{
12                 if(p%i==0){
13                     c++;
14                 }
15                 i++;
16             }while(i<=p && p>=2);
17
18             if(c==2){
19                 printf("%i Es primo",p);
20             }
21             else{
22                 printf("%i No es primo",p);
23             }
24         }
25     }
26     return 0;
27 }
28

length: 33 Ln: 6 Col: 11 Sel: 0|0 Windows (CR LF) UTF-8 INS
```

```
amara@DESKTOP-3AT0670 ~/de19
$ gcc primodo.c -o do

amara@DESKTOP-3AT0670 ~/de19
$ ./do
5
5 Es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
1
no es numero primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
10
10 No es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
11
11 Es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
100
100 No es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
97
97 Es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
88
88 No es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
27
27 No es primo
amara@DESKTOP-3AT0670 ~/de19
$ ./do
9
9 No es primo
amara@DESKTOP-3AT0670 ~/de19
$ |
```

Con la aplicación de estas iteraciones se pudo ver el funcionamiento de cada una de estas, y como varían de una a otra según el caso, pero que cualquiera de estas puede hacer la misma función de repetir un proceso hasta que cumpla una condición.