



NOMBRE DE LA PRÁCTICA	INTRODUCCIÓN AL LENGUAJE C.			NO.	1
ASIGNATURA:	MÉTODOS NUMÉRICOS.	CARRERA:	INGENIERÍA EN SISTEMAS COMPUTACIONALES	DURACIÓN DE LA PRÁCTICA (HRS)	2 HORAS

NOMBRE DEL ALUMNO: Roxana Rubí Miranda Cruz

GRUPO: 3401

I. Competencia(s) específica(s):

- Aplica los tipos de errores para identificar la incertidumbre y limitaciones de los cálculos numéricos en una computadora.

Encuadre con CACEI: Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en esta práctica.

No. atributo	Atributos de egreso del PE que impactan en la asignatura
1	Reconocer la necesidad permanente de conocimiento adicional y tener la habilidad para localizar, evaluar, integrar y aplicar este conocimiento adecuadamente.

II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):

- casa

III. Material empleado:

- Equipo personal

IV. Desarrollo de la práctica:

1 Hola Mundo

Creación del primer programa imprimiendo el mensaje "Hola Mundo"

```

EXPLORADOR  ...  C Holq_Mundo.c X
v RUBI [WSL: UBUNTU]
  > .cache
  > .dotnet
  > .vscode
  > .vscode-server
  > output
  v practicas
    2
    2c

practicas > C Holq_Mundo.c > ...
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      printf("Hola mundo");
5      system("pause");
6      return 0;
7  }
8

```

Ejecución del código

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

● rubi@PC1:~$ cd practicas
✓ rubi@PC1:~/practicas$ gcc Holq_Mundo.c -o Holq_Mundo.out
● rubi@PC1:~/practicas$ ./Holq_Mundo.out
sh: 1: pause: not found
○ Hola mundorubi@PC1:~/practicas$
```

Programa 2 Impresión de Mensajes

Imprimir 2 mensajes en pantalla

Código

```
EXPLORADOR  ...  C Practica2.c X

✓ RUBI [WSL: UBUNTU]
  > .vscode-server
  > output
  ✓ practicas
    ≡ 2
    C 2.c
    C 20.c
    ≡ archivo
    C archivo.c
    C Arreglo.c
    ≡ Arreglo.out

practicas > C Practica2.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      printf("Este es un Texto \n");
5      printf("Fue escrito por Rubi en C \n");
6      printf("\n");
7      printf("Ya soy programador");
8      return 0;
9  }
```

Ejecución de código

```
● rubi@PC1:~/practicas$ gcc Practica2.c -o Practica2.out
● rubi@PC1:~/practicas$ ./Practica2.out
Este es un Texto
Fue escrito por Rubi en C

○ Ya soy programadorrubi@PC1:~/practicas$
```

Programa 3 Programa que imprima la suma de 2 números decimales



Ejecución de Código

```
● rubi@PC1:~/practicass$ ./Practica3.out
3
7
La suma de 3+4 es =7
○ rubi@PC1:~/practicass$
```

File & Run Compile Debug

4 Programa por el cual se calculan la suma de dos números operaciones aritméticas Código

```
EXPLORADOR  ...  C Practica2.c  C Practica3.c  C Practica4.c X
RUBI [WSL: UBUNTU]
  75d1b2ed.de05bb98.crl
  7462804f.d5a68194.crl
  d16da424.cbeb964c.crl
  > .vscode
  > .vscode-server
  > output
  > practicas
    2
    2.c
    20.c
    archivo
    archivo.c
practicass > C Practica4.c > main()
6 //Imprimir el numero 8
7 printf("%d\n",8);
8 printf("La suma de %d+%d es =%d\n",5,8,5+8);
9 //Imprecision del numero 78787
10 printf("%d\n",78787);
11 //Imprecision de el numero 3259
12 printf("%d\n",3259);
13 //Hacer la suma
14 printf("La suma de %d+%d es=%d\n",78787,3259,78787+3259);
15 return 0;
16
17 }
```

Ejecución de código

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL
● rubi@PC1:~$ cd practicas
● rubi@PC1:~/practicass$ gcc Practica4.c -o Practica4.out
● rubi@PC1:~/practicass$ ./Practica4.out
5
8
La suma de 5+8 es =13
78787
3259
La suma de 78787+3259 es=82046
○ rubi@PC1:~/practicass$
```



5 Declaración de variables en C de tipo entero

Código

```
EXPLORADOR  ...  C Practica2.c  C Practica3.c  C Practica4.c  C Practica5.c X
RUBI [WSL: UBUNTU]
7abcc12f.dea36fd7.crl
75d1b2ed.de05bb98.crl
7462804f.d5a68194.crl
d16da424.cbeb964c.crl
> .vscode
> .vscode-server
> output
practicas
2
2.c
20.c

practicas > C Practica5.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int numero1=3;
5      int numero2=4;
6      //imprimir la suma de los dos numeros 3+4
7      printf("La suma de los dos numeros es :");
8      printf("%d", numero1+numero2);
9      printf("\n");
10     return 0;
11 }
```

Ejecución de código

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

● rubi@PC1:~$ cd practicas
● rubi@PC1:~/practicas$ gcc Practica5.c -o Practica5.out
● rubi@PC1:~/practicas$ ./Practica5.out
  La suma de los dos numeros es :7
○ rubi@PC1:~/practicas$
```

6 Programa por el cual calculamos los segundos de vida de acuerdo a nuestra edad

Código

```
EXPLORADOR  ...  C Practica2.c  C Practica3.c  C Practica4.c  C Practica5.c  C Practica6.c
RUBI [WSL: UBUNTU]
meses.c
Practica.out
C Practica2.c
Practica2.out
C Practica3.c
Practica3.out
C Practica4.c
Practica4.out
C Practica5.c
Practica5.out
C Practica6.c
Practica6.out

practicas > C Practica6.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int min=60;
5      int hora=60;
6      int dia=24;
7      int año=365*19;
8      printf("la cantidad de segundos que he vivido son");
9      printf("%d",año*día*hora*min);
10     printf("\n");
11     return 0;
12 }
```



Ejecución del Código

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL

● rubi@PC1:~$ cd practicas
● rubi@PC1:~/practicas$ gcc Practica6.c -o Practica6.out
● rubi@PC1:~/practicas$ ./Practica6.out
○ la cantidad de segundos que he vivido son 630720000
rubi@PC1:~/practicas$
```

7 Programa por el cual calculamos la multiplicación de tres variables con valor definido tipo entero Código

```
RUBI [WSL: ...] practicas > C Practica7.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int vec1=3;
5      int vec2=4;
6      int vec3=6;
7      printf("El resultado de la multiplicacion es");
8      printf("%d",vec1*vec2*vec3);
9      printf("\n");
10     return 0;
11 }
```

Ejecución de código

```
● rubi@PC1:~/practicas$ gcc Practica7.c -o Practica7.out
● rubi@PC1:~/practicas$ ./Practica7.out
El resultado de la multiplicacion es 72
○ rubi@PC1:~/practicas$
```

8 En este programa hacemos uso de operadores asignándoles valores a cada una de ellos Código

```
EXPLORADOR  ...  C Practica8.c X
RUBI [WSL: UBUNTU]
C Practica3.c
Practica3.out
C Practica4.c
Practica4.out
C Practica5.c
Practica5.out
C Practica6.c
Practica6.out
C Practica7.c
Practica7.out
C Practica8.c
Practica8.out
suma.c

practicas > C Practica8.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int x=15;
5      int y=10;
6      int z=++x*y--;
7      printf("x:%d\n",x);
8      printf("y:%d\n",y);
9      printf("z:%d\n",z);
10     return 0;
11 }
```




Ejecución de código

```
rubi@PC1:~/practicass$ gcc Practica8.c -o Practica8.out
rubi@PC1:~/practicass$ ./Practica8.out
x:16
y:9
z:160
rubi@PC1:~/practicass$
```

9 Impresión de las tablas de verdad Código

```
RUBI [WSL: UBUNTU]
C Practica4.c
Practica4.out
C Practica5.c
Practica5.out
C Practica6.c
Practica6.out
C Practica7.c
Practica7.out
C Practica8.c
Practica8.out
C Practica9.c
suma.c
Triangulos.c
.bash_history
.bash_logout
.bashrc
.motd_shown

practicass > C Practica9.c > main()
1 #include <stdio.h>
2 #include <stdlib.h>
3 int main(){
4     printf("-----AND-----\n");
5     printf("true && true : %d\n", (1&&1));
6     printf("true && false : %d\n", (1&&0));
7     printf("false && true : %d\n", (0&&1));
8     printf("false && false : %d\n", (0&&0));
9     printf("-----OR-----\n");
10    printf("true || true : %d\n", (1||1));
11    printf("true || false : %d\n", (1||0));
12    printf("false || true : %d\n", (0||1));
13    printf("false || false : %d\n", (0||0));
14    printf("-----XOR-----\n");
15    printf("true ^ true : %d\n", (1^1));
16    printf("true ^ false : %d\n", (1^0));
17    printf("false ^ true : %d\n", (0^1));
18    printf("false ^ false : %d\n", (0^0));
19    return 0;
20 }
```

Ejecución de código

```
rubi@PC1:~/practicass$ gcc Practica9.c -o Practica.out
rubi@PC1:~/practicass$ ./Practica.out
-----AND-----
true && true : 1
true && false : 0
false && true : 0
false && false : 0
-----OR-----
true || true : 1
true || false : 1
false || true : 1
false || false : 0
-----XOR-----
true ^ true : 0
true ^ false : 1
false ^ true : 1
false ^ false : 0
rubi@PC1:~/practicass$
```



10 Codificación de las tablas de verdad resueltas

Código

```
EXPLORADOR  ...  C Practica8.c  C Practica10.c  C Practica9.c
RUBI [WSL: UBUNTU]  practicas > C Practica10.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int w=9;
5      int x=3;
6      int y=7;
7      int z=-2;
8      printf("Enunciado uno %d\n",((x<y)&(w>z)));
9      printf("Enunciado 2 %d\n",((x>=w)^(z=y)));
10     printf("Enunciado 3 %d\n",((y<=x)^(x!=w)));
11     printf("Enunciado 4 %d\n",((w==9)^(x==3)));
12     printf("Enunciado 5 %d\n",((y<z)&(z=x)));
13     printf("Enunciado 6 %d\n",(!w!=9));
14     return 0;
15 }
```

Ejecución de código

```
● rubi@PC1:~/practicass$ gcc Practica10.c -o Practica10.out
● rubi@PC1:~/practicass$ ./Practica10.out
Enunciado uno 1
Enunciado dos 7
Enunciado tres 1
Enunciado cuatro 0
Enunciado cinco 0
Enunciado seis 1
○ rubi@PC1:~/practicass$
```

11 Programa por el cual codificamos nuestras variables aritméticas

Código

```
C Practica11.c X
practicass > C Practica11.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int x=5;
5      int y=10;
6      int z=++x*y--;
7      printf("x:%d\n",x);
8      printf("y:%d\n",y);
9      printf("x:%d\n",z);
10     return 0;
11 }
```

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```
● rubi@PC1:~$ cd practicass
● rubi@PC1:~/practicass$ gcc Practica11.c -o Practica11.out
● rubi@PC1:~/practicass$ ./Practica11.out
x:6
y:9
x:60
○ rubi@PC1:~/practicass$
```



12 Implementación de operadores lógicos Código

```
EXPLORADOR
RUBI [WSL: UBUNTU]
  Practica6.out
  Practica7.c
  Practica7.out
  Practica8.c
  Practica8.out
  Practica9.c
  Practica10.c
  Practica10.out
  Practica11.c
  Practica11.out
  Practica12.c
  suma.c
  Triangulos.c
  .bash_history
  .bash_logout
  .bashrc
  .motd_shown
  .profile
  .sudo_as_admin_succe...

practicas > C Practica12.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <math.h>
4  int main(){
5      double hip=100;
6      double ang=45;
7      double cateto=45;
8      double cat_opuesto;
9      double cat_ayasen;
10     double at;
11     double ac;
12     cat_opuesto=sin(ang*PI/180)*100;
13     printf("Cateto opuesto =%3f\n",cat_opuesto);
14     cat_ayasen=cos(ang*PI/180)*100;
15     printf("Cateto ayacente = %3f\n",cat_ayasen);
16     at=((cat_opuesto*cat_ayasen)/2);
17     printf("El area del triangulo es %3f\n",at);
18     ac=(at/2);
19     printf("El area del cuadrado es %3f\n",ac);
20     return 0;
21 }
```

13 Resolución de una fracción con variables de valor doblé Código

```
RUBI [WSL: UBUNTU]
  Practica7.c
  Practica7.out
  Practica8.c
  Practica8.out
  Practica9.c
  Practica10.c
  Practica10.out
  Practica11.c
  Practica11.out
  Practica12.c
  Practica13.c
  Practica13.out

practicas > C Practica13.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      double a=1;
5      double b=3;
6      double c=5;
7      double d=30;
8      double e=23;
9      double z;
10     z=((a/b)+(b/c)+(a/d)/(e/d));
11     printf("Resultado %f\n",z);
12     return 0;
13 }
14
```




Ejecución de código

```
● rubi@PC1:~/practicass$ gcc Practica13.c -o Practica13.out
● rubi@PC1:~/practicass$ ./Practica13.out
Resultado 0.976812
○ rubi@PC1:~/practicass$
```

14 Programa por el cual se calcula el área de un triángulo

Código

```
EXPLORADOR  ...  C Practica11.c  C Practica13.c  C Practica14.c  C Practica15.c X
RUBI [WSL: UBUNTU]
Practica8.out
C Practica9.c
C Practica10.c
Practica10.out
C Practica11.c
Practica11.out
C Practica12.c
C Practica13.c
Practica13.out
C Practica14.c
Practica14.out
Practica15
C Practica15.c
Practica15.out

practicass > C Practica15.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4  //Evaluar una calificacion
5  int numero;
6  printf("Escribe la calificacion");
7  scanf("%d",&numero);
8  if(numero >=7){
9  printf("Tu calificacion es aprobatoria");
10 }else
11 printf("Calificacion minima aprobatoria es 8");
12 return 0;
13 }
```

Ejecución de código

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL
● rubi@PC1:~$ cd practicas
● rubi@PC1:~/practicass$ gcc Practica15.c -o Practica15.out
● rubi@PC1:~/practicass$ ./Practica15.out
Escribe la calificacion8
○ Tu calificacion es aprobatoria
```

15 Programa para Calcular si un número es impar o para

Código

```
RUBI [WSL: UBUNTU]
C Practica8.c
Practica8.out
C Practica9.c
C Practica10.c
Practica10.out
C Practica11.c
Practica11.out
C Practica12.c
C Practica13.c
Practica13.out
C Practica14.c

practicass > C Practica14.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4  //numeros pares e impares
5  int numero=55;
6  if(numero%2==0){
7  printf("El numero es par \n");
8  }
9  printf("***Fin del programa***");
10 return 0;
11 }
```



```
● rubi@PC1:~/practicas$ gcc Practica14.c -o Practica14.out
● rubi@PC1:~/practicas$ ./Practica14.out
○ **Fin del programa**rubi@PC1:~/practicas$
```

Código

Ejecución de código

```
PROBLEMAS      SALIDA      CONSOLA DE DEPURACIÓN      TERMINAL
```

```
● rubi@PC1:~$ cd practicas
● rubi@PC1:~/practicas$ gcc Practica15.c -o Practica15.out
● rubi@PC1:~/practicas$ ./Practica15.out
Escribe la calificacion8
○ Tu calificacion es aprobatoria● rubi@PC1:~/practicas$
```

Código

```
RUBI [WSL: UBUNTU]
practicas > C Practica17.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int hst;
5      int sse;
6      int ph;
7      int ss;
8      printf("Horas semanales trabajadas:\n");
9      scanf("%d",&hst);
10     printf("Pago por hora \n");
11     scanf("%d",&ph);
12     ss=(hst*ph);
13     if(hst<=40){
14         printf("El sueldo semanal es : $ %d\n",ss);
15     }else if(hst>40){
16         sse=(1.5*(hst-40)*ph);
17         printf("El sueldo extra es:$ %d\n",sse);
18         ss=(40*ph+(ph*1.5*(hst-40)));
19         printf("sueldo total semanal con extra es de :$ %d\n",ss);
20     }
21     return 0;
22 }
23
```



Ejecución de código

```

rubi@PC1:~/practicass$ gcc Practica17.c -o Practica17.out
rubi@PC1:~/practicass$ ./Practica17.out
Horas semanales trabajadas:
6
Pago por hora
7
El sueldo semanal es : $ 42
rubi@PC1:~/practicass$

```

18 Programa en el cual se escriba un número y se calcule números de asteriscos Código

EXPLORADOR ... C Practica11.c C Practica16.c C Practica17.c C Triangulos.c

RUBI [WSL: UBUNTU]

```

practicass > C Triangulos.c > main(void)
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main(void)
5  {
6      int lineas;
7      printf("Ingresa la cantidad a la que desea llagar \n ");
8      scanf("%d", &lineas);
9      int i;
10     for (i = lineas; i >= 1; i--)
11     {
12         int j;
13         for (j = 0; j < i; j++)
14         {
15             printf("*");
16         }
17         printf("\n");
18     }
19     return 0;
20 }
21

```

Ejecución de código

PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL

```

rubi@PC1:~$ cd practicas
rubi@PC1:~/practicass$ gcc Triangulos.c -o Triangulos.out
rubi@PC1:~/practicass$ ./Triangulos.out
Ingresa la cantidad a la que desea llagar
6
*****
*****
*****
*****
*****
*****
*

```




21 Calcula los números divisibles entre 3 mayor a 0 y menor 1000

Código

```
practicas > C Practica19.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4      int num =3;
5      while(num<1000){
6          if(num%3==0){
7              printf("%d",num);
8              printf("\n");
9          }
10         num +=3;
11     }return 0;
12 }
13
```

Ejecución de código

```
● rubi@PC1:~$ cd practicas
● rubi@PC1:~/practicas$ gcc Practica19.c -o Practica19
● rubi@PC1:~/practicas$ ./Practica19
3
6
9      78
12     81
15     84      915
18     87      918
21     90      921
24     93      924
27     96      927
30    102      930
33    105      933
36    108      936
39    111      939
42    114      942
45    117      945
48    120      948
51    123      951
54    126      954
57    129      957
60    132      960
63    135      963
66    138      966
69    141      969
72    144      972
75    147      975
78    150      978
81    153      981
      156      984
      159      987
      162      990
      165      993
              996
              999
● rubi@PC1:~/practicas$
```




22 Arreglos

Imprimir un arreglo de 10 elementos sin usar ciclos

Código

```
practicas > Practica20.c > main()
1  #include <stdio.h>
2  #include <stdlib.h>
3  int main(){
4
5      int miarreglo[10];
6      miarreglo[0]=10;
7      miarreglo[1]=10;
8      miarreglo[2]=10;
9      miarreglo[3]=10;
10     miarreglo[4]=10;
11     miarreglo[5]=10;
12     miarreglo[6]=10;
13     miarreglo[7]=10;
14     miarreglo[8]=10;
15     miarreglo[9]=10;
16
17     printf("%d",miarreglo[0]);
18     printf("%d",miarreglo[1]);
19     printf("%d",miarreglo[2]);
20     printf("%d",miarreglo[3]);
21     printf("%d",miarreglo[4]);
22     printf("%d",miarreglo[5]);
23     printf("%d",miarreglo[6]);
24     printf("%d",miarreglo[7]);
25     printf("%d",miarreglo[8]);
26
```

Ejecución de código



```
• rubi@PC1:~/practicass$ gcc Practica20.c -o Practica20.out
• rubi@PC1:~/practicass$ ./Practica20.out
• 1010101010101010rubi@PC1:~/practicass$ █
```

Conclusiones

La conclusión a la que se llegó es que hay que reforzar los conocimientos para poder implementar y desarrollar programas en lenguaje C además de que con cada práctica se aumenta la probabilidad de el desarrollo de las habilidades necesarias que se requieren para cumplir con lo que demanda la asignatura de Métodos Numéricos