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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Cruz Carlon Juan Alfredo |
| *Asignatura:* | Fundamentos de programación |
| *Grupo:* | 1107 |
| *No de Práctica(s):* | 10 “Depuración de Programas”. |
| *Integrante(s):* | Fabila Zuñiga Miguel Angel |
|  | Rodriguez Aguilar Sergio Gustavo |
|  | Soria Pérez Escoly Nanui |
| *Semestre:* | 2018-1 |
| *Fecha de entrega:* | 12 de noviembre de 2017 |
| *Obervaciones:* |  |
| CALIFICACIÓN: |  |

Fundamentos de Programacio´n Ejercicios de depuracio´n.

# Llena la tabla para cada uno de los ejercicios.

|  |  |  |  |
| --- | --- | --- | --- |
| Programa | Lo que deber´ıa realizar | Lo que realiza | Reparaci´on |
| **#include** *<* s t d i o . h*>*  **int** main ( )  *{*  **int** r e s p u e s t a ; r e s p u e s t a = 2 + 2 ;  p r i n t f ( ” La r e s p u e s t a es : %d*\*n” ) ;  **return** 0 ;  *}* | Imprimir **La respuesta es: 4** | Imrpime la respuesta es: -401867848 | Agregandole en el printf despues de %d”, respuesta); |
| **#include** *<* s t d i o . h*>*  **int** main ( )  *{*  **f loat** r e s p u e s t a ; r e s p u e s t a = 7 . 0 / 2 2 . 0 ;  p r i n t f ( ” La r e s p u e s t a es : %d*\*n” , r e s p u e s t a ) ;  **return** 0 ;  *}* | Imprimir **La respuesta es: 0.3181818** | Imprime la respuesta es: 387689960 | Cambiamos el %d por un %f ya que con el %d solo nos referimos a los enteros y con el %f a los reales |
| **#include** *<* s t d i o . h*>*  **int** main ( )  *{*  **char** l i n e a [ 8 0 ] ;  **int** balance ;  p r i n t f ( ” In gr esa e l balance ( en pesos ) de l a cuenta : ” ) ;  f g e t s ( l i n e a , **sizeof** ( l i n e a ) , s t d i n ) ; s s c a n f ( l i n e a , ” %d” , &balance ) ;  **i f** ( balance = 0 ) *{*  p r i n t f ( ”No t i e n e s dinero *\*n” ) ;  *}* **else** *{*  p r i n t f ( ” Tienes : %d pesos . *\* n” , balance ) ;  *}*  **return** 0 ;  *}* | Imprimir el balance ingresado | Cuando ingresamos cualquier valor imprime tienes 0 pesos. | Solo hay que modificar la condición if modificándola como if(<=0) |

1

2

3

4

5

6

7

8

9

1

2

3

4

5

6

7

8

9

1

2

3

4

5

6

7

8

9

10

11

12

13

14

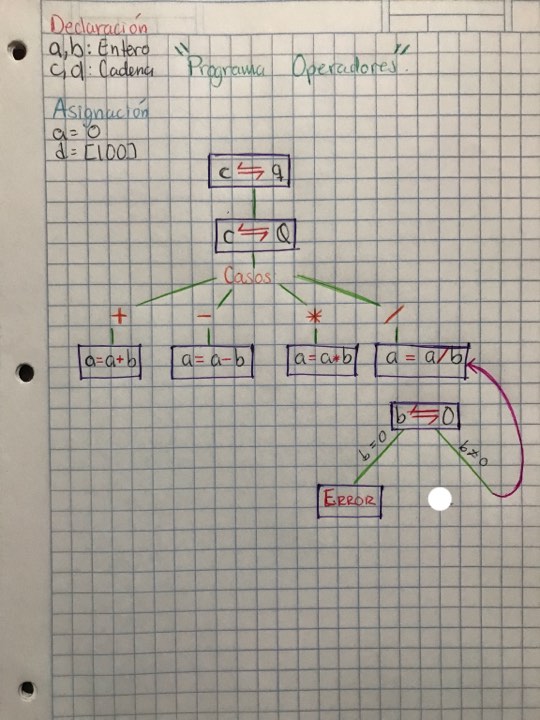
15

16

17

REPRESENTACION GRAFICA DEL CODIGO

# Determina que *hace* el siguiente programa. Explica en que contribuye cada línea y haz una representación gráfica del mismo.



# CODIGO EN LENGUAJE C

# 1 **#include** *<* s t d i o . h*> //indica la librería que se va a utilizar*

2

3 **int** main ( ) //indica el inicio del programa

4 { //indica el inicio del código

5 **int** a ; //declara la variable de tipo entero

6 **int** b ; //declara la variable de tipo entero

7 **char** c ; //char es para una cadena de texto y se declarara la variable c

8 **char** d [ 1 0 0 ] ; //indica el tamaño de la cadena

9 a = 0 ; //se asigna un valor a la variable

10

11 **while** ( 1 ) *{ //indica que va a comenzar un ciclo*

12 p r i n t f ( ”A: %d*\*n” , a ) ; //se escribirá A y su valor es el de a o sea 0

13

14 p r i n t f ( ” Operador ? : ” ) ; //se escribirá en como pregunta operador

15 f g e t s ( d , **sizeof** ( d ) , s t d i n ) ;//dice que es una cadena y se lee

16 s s c a n f ( d , ” %c” , &c ) ;//se lee el valor de la variable c

17

18 p r i n t f ( ” Numero ? : ” ) ; //se imprime y se pregunta el numero

19 f g e t s ( d , **sizeof** ( d ) , s t d i n ) ;//dice que es una cadena y la lee

20 s s c a n f ( d , ” %d” , &b ) ;// se lee el valor de la variable b

21

22 **i f** ( c == ’ q ’ )// es una condición

23 **break** ;//detiene la ejecución de la condición y cambia a otra

24 **i f** ( c == ’Q’ )//se declara otra condición

25 **break** ;// se detiene la ejecución de la condición y cambia a otra

26 **switch** ( c )//hace un cambio de instrucción a la siguiente

*{*

27 **case** ’+’ : //introduce que puede haber otro caso

28 a = a + b ;// se asigna una expresión a la variable a

29 **break** ; // se detiene la ejecución de la condición y cambia a otra

30 **case** ’ ’ :b//dice que hacer en otro caso

*−*

31 a = a b ;//se asigna una expresión a la variable a

*−*

32 **break** ; // se detiene la ejecución de la condición y cambia a otra

33 **case** ’ ’ :

*∗*

34 a = a b ;//se asigna una expresión a la variable a

*∗*

35 **break** ; // se detiene la ejecución de la condición y cambia a otra

36 **case** ’ / ’ ://dice que hacer en otro caso

37 **i f** ( b == 0 )//indica otra condición

*{*

38 p r i n t f ( ” Error n” ) ;//imprime que hay un error

*\*

39 **else// indica que hacer en caso contrario**

*} {*

40 a = a/b ;//se asigna una expresión a la variable a

41

*}*

42 **break** ; // se detiene la ejecución de la condición y cambia a otra

43 **default** ://guarda los datos que se piden

44 p r i n t f ( ”Op. desconocida %c n” , c ) ;//se escribe operación desconocida

*\*

45 **break** ; // se detiene la ejecución de la condición y cambia a otra

46 *}*

47

48

*}*

49 **return** 0 ;//fin de las operaciones

50 *}//fin del programa*

EXPLICACION DEL PROGRAMA.

Lo que realiza el programa es preguntar por un operador que puede ser suma, resta, multiplicación o la división, también pregunta por los números para efectuar alguna operación. Depende del operador que ingrese el usuario realizara una función como a=a+b, a=a-b, a=a\*b, a=a/b.