Fundamentos de Programaci´on Ejercicios de depuración.

# 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*>*  **i n t** main ( )  {  **i n t** 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 e s : %d\n” ) ;  **return** 0 ;  } | Imprimir **La respuesta es: 4** | La respuesta es: -401867848 | Modificando: printf("La respuesta es:%d",respuesta); |
| **#include** *<* s t d i o . h*>*  **i n t** main ( )  {  **f l o a t** 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 e s : %d\n” , r e s p u e s t a ) ;  **return** 0 ;  } | Imprimir **La respuesta es: 0.3181818** | La respuesta es: 387689960 | Cambiarle el %d  Por el %f |
| **#include** *<* s t d i o . h*>*  **i n t** main ( )  {  **char** l i n e a [ 8 0 ] ;  **i n t** b a l a n c e ;  p r i n t f ( ” I n g r e s a e l b a l a n c e ( en p e s o s ) de l a c u e n t a : ” ) ;  f g e t s ( l i n e a , **s i z e o f** ( l i n e a ) , s t d i n ) ; s s c a n f ( l i n e a , ” %d” , &b a l a n c e ) ;  **i f** ( b a l a n c e = 0 ) {  p r i n t f ( ”No t i e n e s d i n e r o \n” ) ;  } **e l s e** {  p r i n t f ( ” T i e n e s : %d p e s o s . \ n” , b a l a n c e ) ;  }  **return** 0 ;  } | Imprimir el balance ingresado | no realizaba esa indicación | En:  If(balance=0){  Le agregue otro = |

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

# Determina que *hace* el siguiente programa. Explica en qu´e contribuye cada l´ınea y haz una representaci´on gr´afica del mismo.

1 **#include** *<* s t d i o . h*>*

2

3 **i n t** main ( )

{

4

5 **i n t** a ;

6 **i n t** b ;

7 **char** c ;

8 **char** d [ 1 0 0 ] ;

9 a = 0 ;

10

11 **while** ( 1 ) {

12 p r i n t f ( ”A: %d\n” , a ) ;

13

14 p r i n t f ( ” Operador ? : ” ) ;

15 f g e t s ( d , **s i z e o f** ( d ) , s t d i n ) ;

16 s s c a n f ( d , ” %c ” , &c ) ;

17

18 p r i n t f ( ”Numero ? : ” ) ;

19 f g e t s ( d , **s i z e o f** ( d ) , s t d i n ) ;

20 s s c a n f ( d , ” %d” , &b ) ;

21

22 **i f** ( c == ’ q ’ )

23 **break** ;

24 **i f** ( c == ’Q ’ )

25 **break** ;

26 **switch** ( c )

{

27 **case** ’+ ’ :

28 a = a + b ;

29 **break** ;

30 **case** ’ ’ :

−

31 a = a b ;

−

32 **break** ;

33 **case** ’ ’ :

∗

34 a = a b ;

∗

35 **break** ;

36 **case** ’ / ’ :

37 **i f** ( b == 0 )

{

38 p r i n t f ( ” E r r o r n” ) ;

\

39 **e l s e**

} {

40 a = a /b ;

}

41

42 **break** ;

43 **d e f a u lt** :

44 p r i n t f ( ”Op . d e s c o n o c i d a %c n” , c ) ;

\

45 **break** ;

46 }

47

48

}

49 **return** 0 ;

50 }