

Brahian Castro Devia

Tarea02

Presentado a: Doctor Ricardo Moreno Laverde

Fecha: 04/03/2024

Universidad Tecnológica De Pereira

Código 1: Identificar si un numero es primo o no

Enlace: <https://youtu.be/TI30RcJ6xzY>

Salida en pantalla:

```
Este programa, lee un número entre uno(1) y cinco(5), e
imprime si es primo o no.
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número: 5
El número es primo
```

```
Este programa, lee un número entre uno(1) y cinco(5), e
imprime si es primo o no.
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número: 2
El número es primo
```

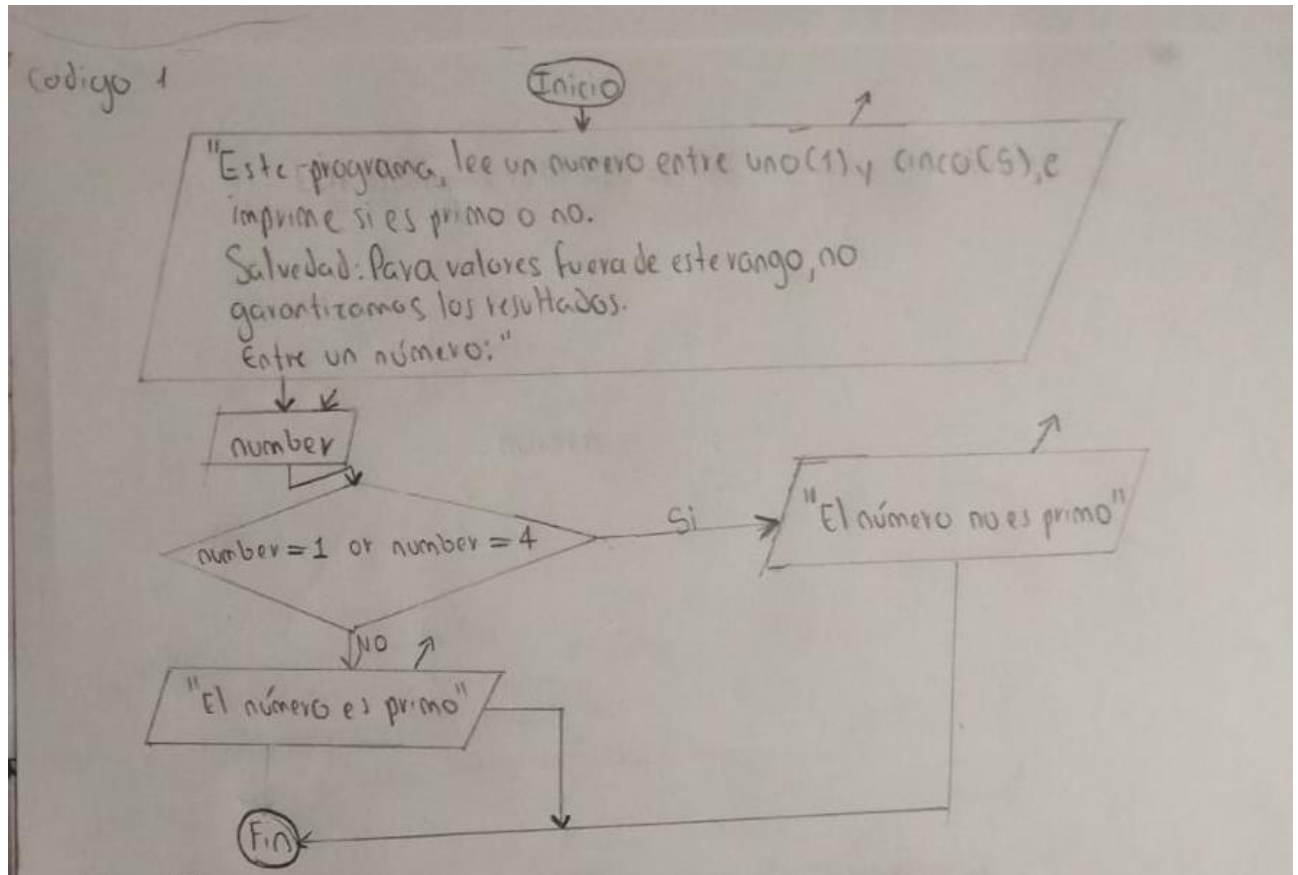
```
Este programa, lee un numero entre uno(1) y cinco(5), e
imprime si es primo o no.
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número: 1
El número no es primo
```

Datos entrada y salida:

Input: 2 (número a dar por el usuario)

Output: El numero es primo

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 19:08

-Versión de su código: 4

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa solicita un numero al usuario , el cual despues analiza y define si este numero es primo o no

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje , para valores fuera del rango de 1 a 5 , no garantizamos resultados

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number se usa para almacenar el numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeNumber)

(printf "Este programa, lee un número entre uno(1) y cinco(5), e
imprime si es primo o no.

Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.

Entre un número: ")

(define number (read))

(if (or (= number 1) (= number 4))

```
(printf "El número no es primo")  
(printf "El número es primo")  
)  
)  
( AnalyzeNumber )
```

Código 2:

Enlace: <https://youtu.be/64XjpgclAlU>

Salida en pantalla:

```
Este programa, lee un número e imprime si es par o  
impar.  
Entre un número: 4  
El número es par
```

```
Este programa, lee un número e imprime si es par o  
impar.  
Entre un número: 533  
El número no es par
```

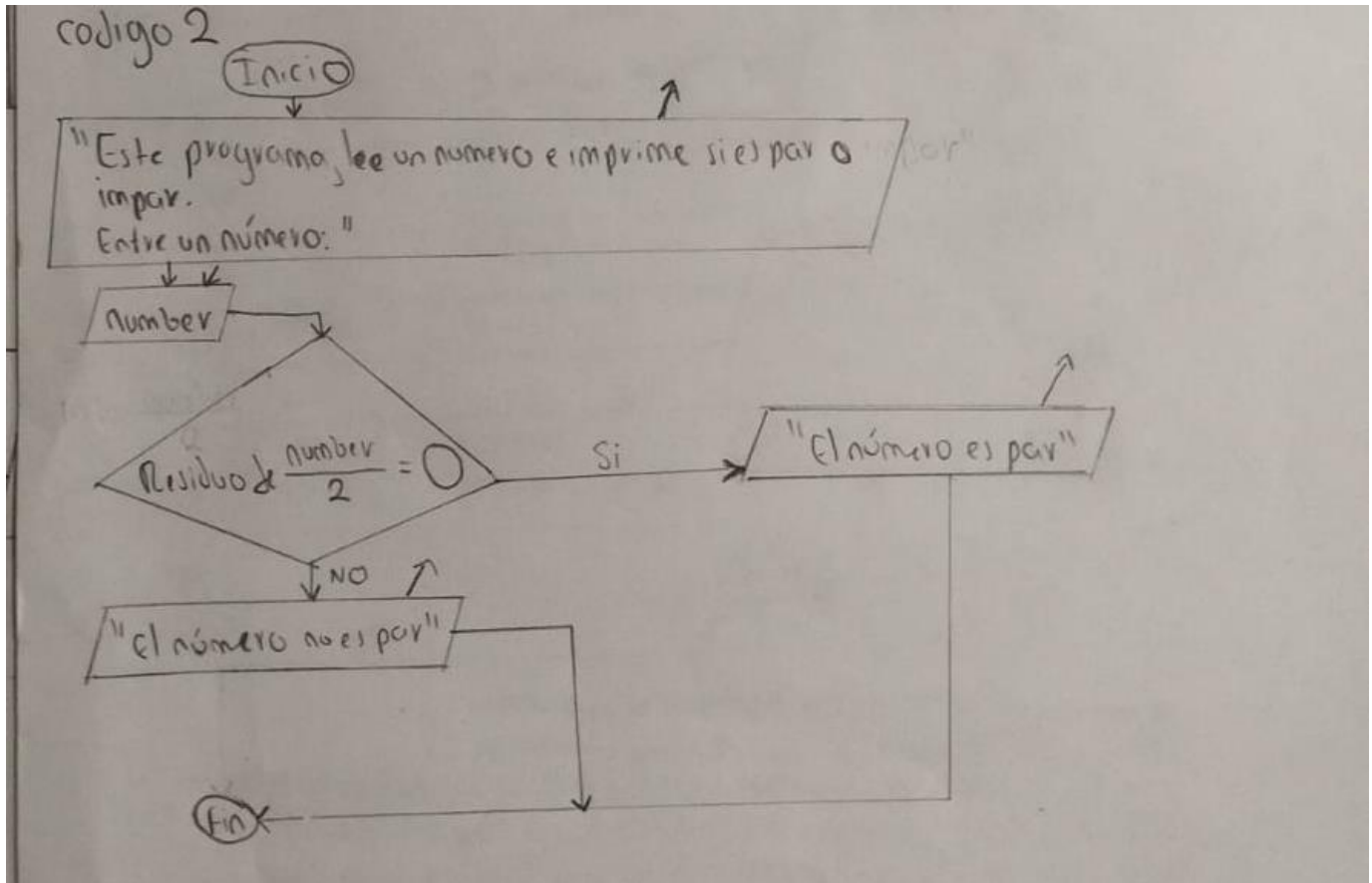
```
Este programa, lee un número e imprime si es par o  
impar.  
Entre un número: 1002945  
El número no es par
```

Datos entrada y salida:

Input: 4 (numero ingresado por el usuario)

Output: El numero es par

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 19:15

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa solicita un numero al usuario , el cual despues analiza y define si este numero es par o no

en base al residuo entre el numero y 2

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje ,

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number se usa para almacenar el numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeNumber)

(printf "Este programa, lee un número e imprime si es par o impar.

Entre un número: ")

(define number (read))

(if (= (remainder number 2) 0)

(printf "El número es par")

(printf "El número no es par")

)

)

(AnalyzeNumber)

Código 3: Que pida un número del 1 al 7 e imprima el día de la semana correspondiente.

Enlace: <https://youtu.be/ycGnv3UyQck>

Salida en pantalla:

```
Este programa, lee un número entre [1 y 7] e imprime el
día que le corresponde en la semana. Referencia:
1=Lunes. 7=Domingo
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número: 4
El día es Jueves
```

```
Este programa, lee un número entre [1 y 7] e imprime el
día que le corresponde en la semana. Referencia:
1=Lunes. 7=Domingo
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número: 1
El día es Lunes
```

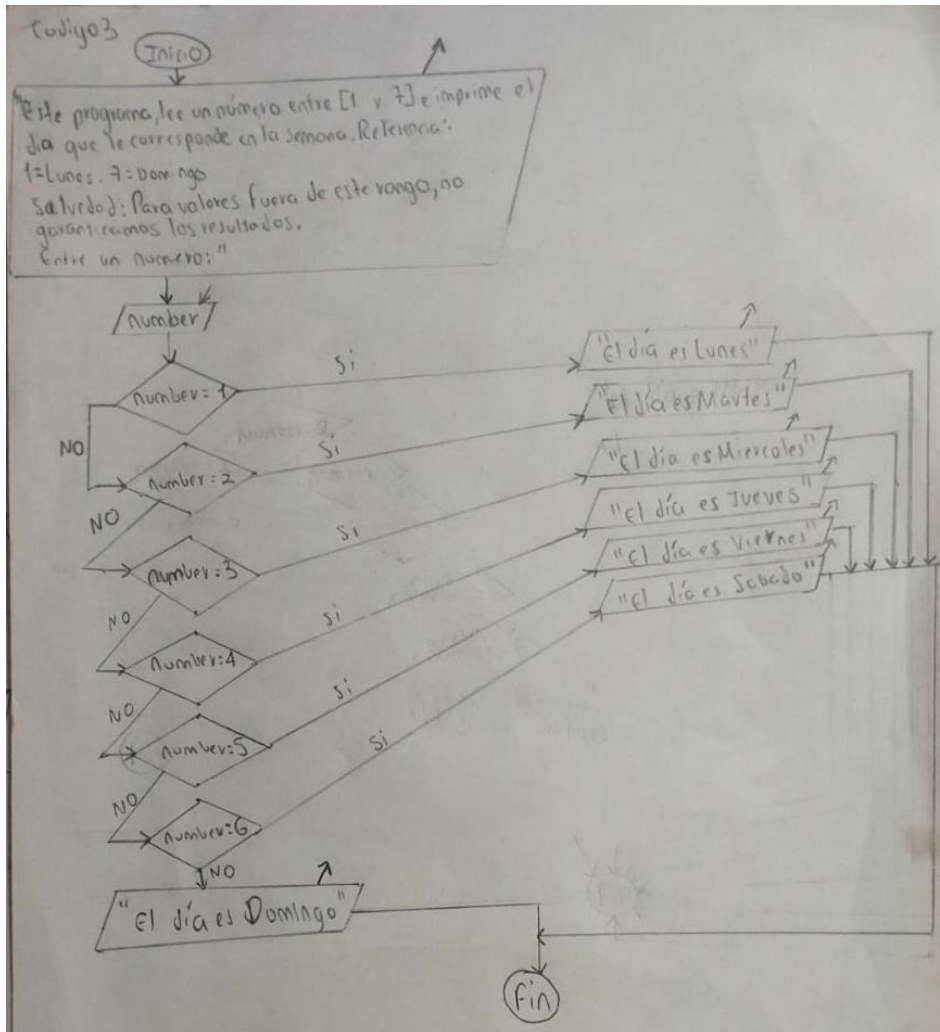
```
Este programa, lee un número entre [1 y 7] e imprime el
día que le corresponde en la semana. Referencia:
1=Lunes. 7=Domingo
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número: 7
El día es Domingo
```

Datos entrada y salida:

Input: 4 (numero ingresado por el usuario)

Output: El día es Jueves

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 19:38

-Versión de su código: 3

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa solicita un numero al usuario , Con el cual imprime en pantalla el dia de la semana el cual esta

representado por este numero

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje , Salvedad: Para valores fuera del rango del 1 al 7 , no

garantizamos los resultados.

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number se usa para almacenar el numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeDay)

(printf "Este programa, lee un número entre [1 y 7] e imprime el día que le corresponde en la semana. Referencia:

1=Lunes. 7=Domingo

Salvedad: Para valores fuera de este rango, no garantizamos los resultados.

```
Entre un número: " )  
( define number (read ) )  
( if (= number 1)  
  (printf"El día es Lunes")  
  ( if (= number 2)  
    (printf"El día es Martes")  
    ( if (= number 3)  
      (printf"El día es Miercoles")  
      ( if (= number 4)  
        (printf"El día es Jueves")  
        ( if (= number 5)  
          (printf"El día es Viernes")  
          ( if (= number 6)  
            (printf"El día es Sabado")  
            (printf"El día es Domingo")  
          )  
        )  
      )  
    )  
  )  
)  
( AnalyzeDay )
```

Código 4: Que pida un número del 1 al 12 e imprima el nombre del mes correspondiente.

Enlace: <https://youtu.be/m5RTVMBxDBU>

Salida en pantalla:

```
Este programa, lee un número entre [1 y 12] e imprime
el mes que le corresponde en el año. Referencia:
1=Enero. 12=Diciembre
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número:4
El mes es Abril
```

```
Este programa, lee un número entre [1 y 12] e imprime
el mes que le corresponde en el año. Referencia:
1=Enero. 12=Diciembre
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número:1
El mes es Enero
```

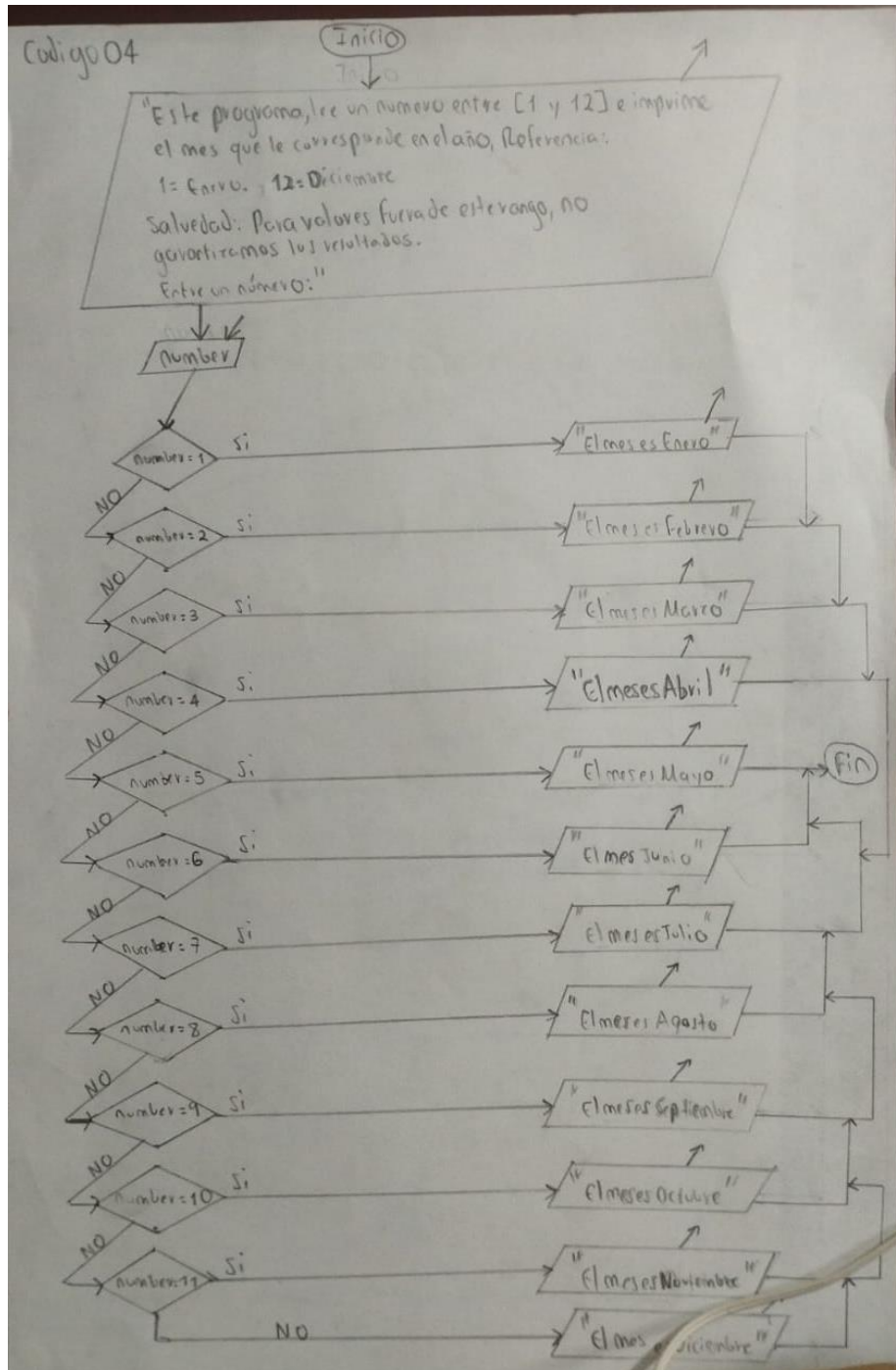
```
Este programa, lee un número entre [1 y 12] e imprime
el mes que le corresponde en el año. Referencia:
1=Enero. 12=Diciembre
Salvedad: Para valores fuera de este rango, no
garantizamos los resultados.
Entre un número:7
El mes es Julio
```

Datos entrada y salida:

Input: 4 (numero ingresado por el usuario)

Output: El mes es Abril

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 20:24

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa solicita un numero en el rango de 1 a 12 ingresado por el usuario y devuelve el mes

con el valor correspondiente al numero

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

- Para valores fuera del rango de 1 y 12 no se garantizan resultados

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number se utiliza para almacenar el numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeMonth)

(printf"Este programa, lee un número entre [1 y 12] e imprime

el mes que le corresponde en el año. Referencia:

1=Enero. 12=Diciembre

Salvedad: Para valores fuera de este rango, no

garantizamos los resultados.

Entre un número:")


```
( define number ( read ))  
  
  ( if ( = number 1 )  
    ( printf"El mes es Enero" )  
    ( if ( = number 2 )  
      ( printf"El mes es Febrero" )  
      ( if ( = number 3 )  
        ( printf"El mes es Marzo" )  
        ( if ( = number 4 )  
          ( printf"El mes es Abril" )  
          ( if ( = number 5 )  
            ( printf"El mes es Mayo" )  
            ( if ( = number 6 )  
              ( printf"El mes es Junio" )  
              ( if ( = number 7 )  
                ( printf"El mes es Julio" )  
                ( if ( = number 8 )  
                  ( printf"El mes es Agosto" )  
                  ( if ( = number 9 )  
                    ( printf"El mes es Septiembre" )  
                    ( if ( = number 10 )  
                      ( printf"El mes es Octubre" )  
                      ( if ( = number 11 )  
                        ( printf"El mes es Noviembre" )  
                        ( printf"El mes es Diciembre" )  
                      )  
                    )  
                  )  
                )  
              )  
            )  
          )  
        )  
      )  
    )  
  )  
)))))))))  
  
( AnalyzeMonth)
```

Código 5: Que leyendo por teclado la antigüedad en años, que calcule el sueldo mensual que le corresponde al

trabajador de una empresa que cobra 40.000 euros anuales, el programa debe realizar los cálculos en función

de los siguientes criterios:

Enlace: <https://youtu.be/7j-BzIm3diA>

Salida en pantalla:

```
Este programa leyendo por teclado la antigüedad en
años, calcula el sueldo mensual que le corresponde al
trabajador de una empresa que cobra 40.000 euros
anuales, el programa
debe realizar los cálculos en función de los
siguientes criterios:
a. Si lleva más de 10 años en la empresa se le aplica
un aumento del 10%.
b. Si lleva menos de 10 años pero más que 5 se le
aplica un aumento del 7%.
c. Si lleva menos de 5 años pero más que 3 se le
aplica un aumento del 5%.
d. Si lleva menos de 3 años se le aplica un aumento
del 3%.
Entre el número de años de antigüedad del trabajador:10
El sueldo mensual es de 3566.6666666666665 euros >
```

```
Este programa leyendo por teclado la antigüedad en
años, calcula el sueldo mensual que le corresponde al
trabajador de una empresa que cobra 40.000 euros
anuales, el programa
debe realizar los cálculos en función de los
siguientes criterios:
a. Si lleva más de 10 años en la empresa se le aplica
un aumento del 10%.
b. Si lleva menos de 10 años pero más que 5 se le
aplica un aumento del 7%.
c. Si lleva menos de 5 años pero más que 3 se le
aplica un aumento del 5%.
d. Si lleva menos de 3 años se le aplica un aumento
del 3%.
Entre el número de años de antigüedad del trabajador:24
El sueldo mensual es de 3666.6666666666665 euros > |
```

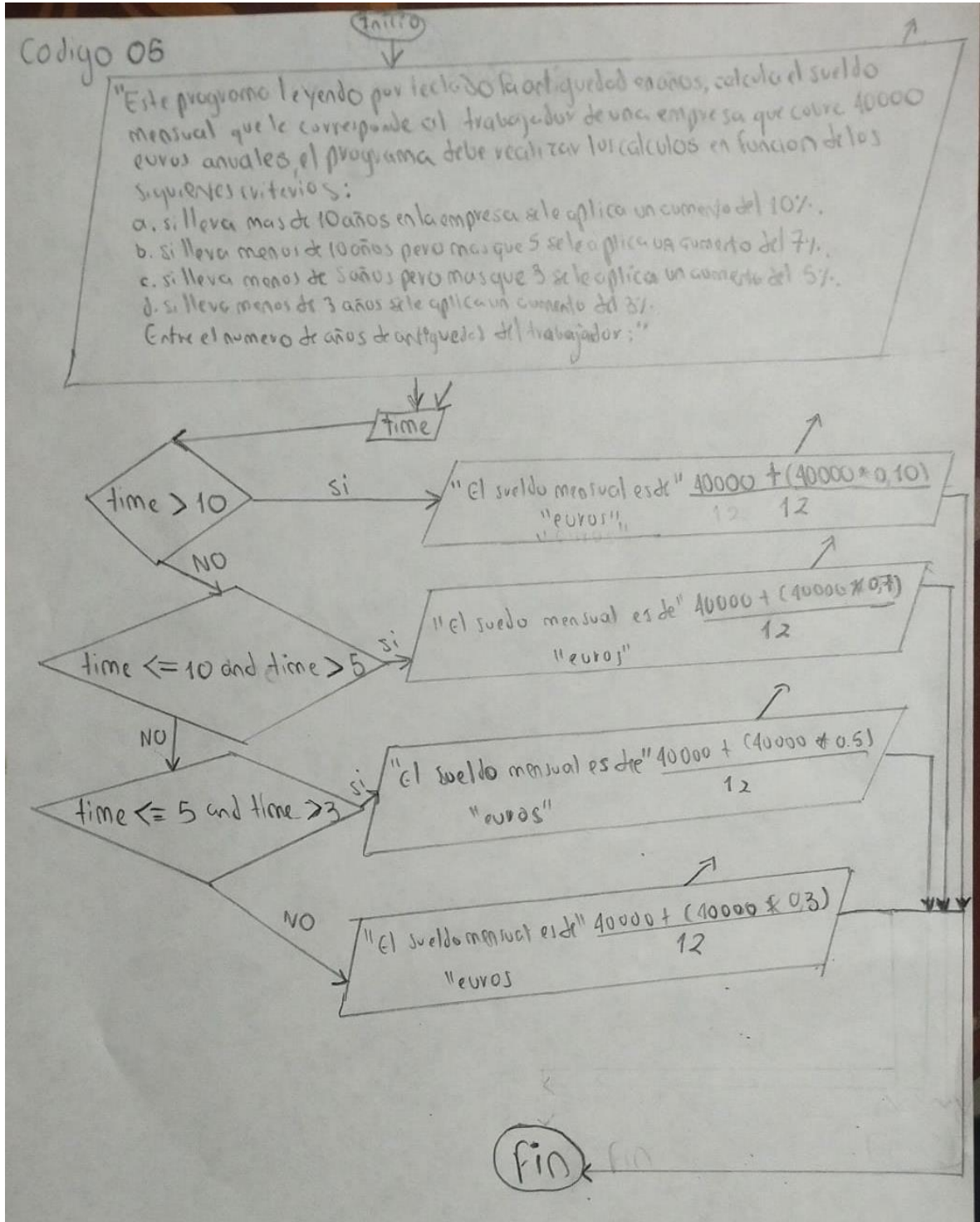
```
Este programa leyendo por teclado la antigüedad en
años, calcula el sueldo mensual que le corresponde al
trabajador de una empresa que cobra 40.000 euros
anuales, el programa
debe realizar los cálculos en función de los
siguientes criterios:
a. Si lleva más de 10 años en la empresa se le aplica
un aumento del 10%.
b. Si lleva menos de 10 años pero más que 5 se le
aplica un aumento del 7%.
c. Si lleva menos de 5 años pero más que 3 se le
aplica un aumento del 5%.
d. Si lleva menos de 3 años se le aplica un aumento
del 3%.
Entre el número de años de antigüedad del trabajador:5
El sueldo mensual es de 3500.0 euros
```

Datos entrada y salida:

Input: 10 (antigüedad ingresada por el usuario)

Output: 3566.666666666665 (sueldo calculado)

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 20:08

-Versión de su código: 5

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa solicita la antigüedad al usuario por medio del teclado , con este dato devuelve el sueldo con

el aumento correspondiente

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : time se utiliza para almacenar el valor de la antigüedad del usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePay)

(printf"Este programa leyendo por teclado la antigüedad en años, calcula el sueldo mensual que le corresponde al

trabajador de una empresa que cobra 40.000 euros anuales, el programa

debe realizar los cálculos en función de los siguientes criterios:

- a. Si lleva más de 10 años en la empresa se le aplica un aumento del 10%.
- b. Si lleva menos de 10 años pero más que 5 se le aplica un aumento del 7%.
- c. Si lleva menos de 5 años pero más que 3 se le aplica un aumento del 5%.

d. Si lleva menos de 3 años se le aplica un aumento del 3%.

Entre el número de años de antigüedad del trabajador:")

```
( define time ( read ) )
```

```
( if ( > time 10 )
```

```
  ( printf " El sueldo mensual es de ~a euros " ( / ( + 40000 ( * ( / 40000 100 ) 10 ) ) 12. ) )
```

```
  ( if ( and( <= time 10 ) ( > time 5 ) )
```

```
    ( printf " El sueldo mensual es de ~a euros " ( / ( + 40000 ( * ( / 40000 100 ) 7 ) ) 12. ) )
```

```
    ( if ( and( <= time 5 ) ( > time 3 ) )
```

```
      ( printf " El sueldo mensual es de ~a euros " ( / ( + 40000 ( * ( / 40000 100 ) 5 ) ) 12. ) )
```

```
      ( printf " El sueldo mensual es de ~a euros " ( / ( + 40000 ( * ( / 40000 100 ) 3 ) ) 12. ) )
```

```
    ) ) ) )
```

```
( AnalyzePay )
```

Código 6: Que pida un número y diga si es mayor de 100.

Enlace: https://youtu.be/o7oRQJ_eluQ

Salida en pantalla:

```
Este programa, Analiza un numero ingresado y determina si es
mayor a 100
Salvedad: Para valores expresados en palabras no se garantizan
resultados.
Entre un número:7999
Es mayor que 100
```

```
Este programa, Analiza un numero ingresado y determina si es
mayor a 100
Entre un número:5
```

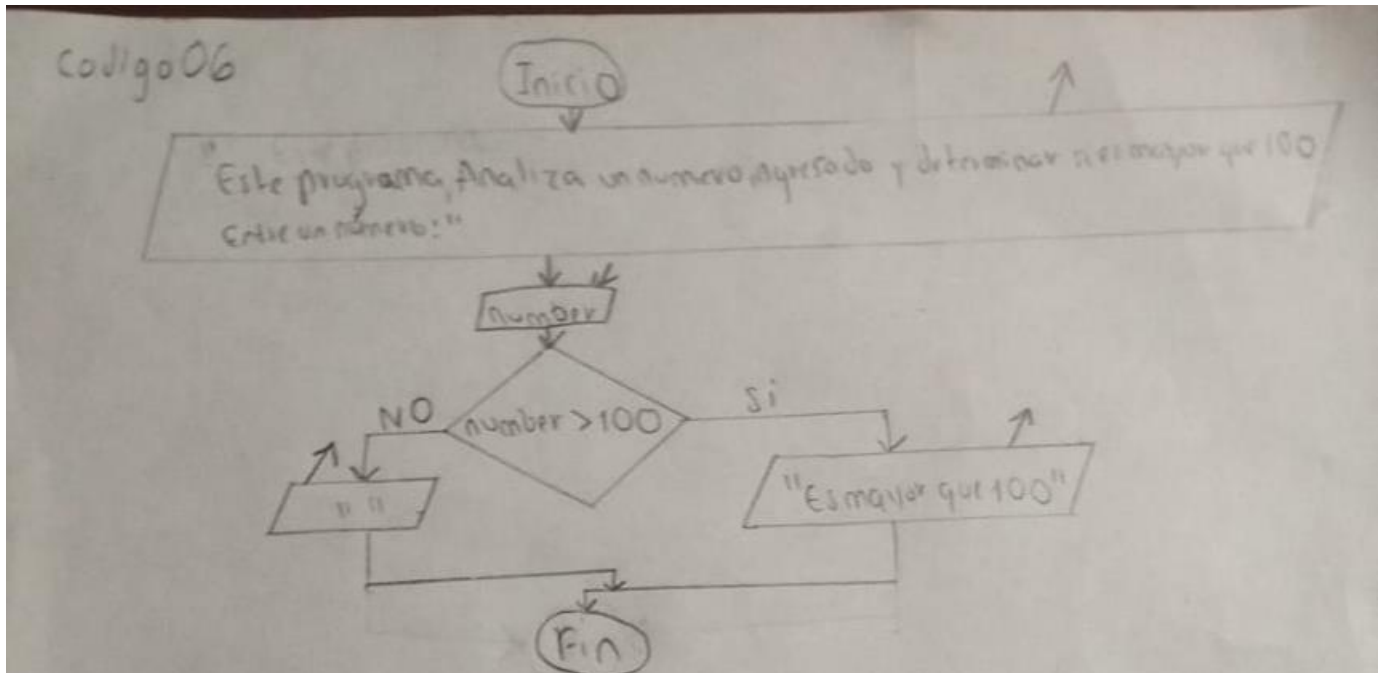
```
Este programa, Analiza un numero ingresado y determina si es
mayor a 100
Entre un número:99
```

Datos entrada y salida:

Input: 7999 (numero ingresado por el usuario)

Output: Es mayor que 100

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 21:53

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa analiza un numero ingresado por el usuario y determina si es mayor a 100

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number se utiliza para almacenar el numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeNumber)

(printf"Este programa, Analiza un numero ingresado y determina si es mayor a 100

Entre un número:")

(define number (read))

(if (> number 100)

(printf "Es mayor que 100")

(printf " ")

))

(AnalyzeNumber)

Código 7: Que tome dos números y diga si ambos son pares o ambos son impares.

Enlace: <https://youtu.be/unBNv6NboKA>

Salida en pantalla:

```
Este programa, Analiza dos numeros ingresados y determina si  
ambos son pares o ambos son impares.  
Entre un número:2  
Entre otro numero4  
Ambos numeros son pares
```

```
Este programa, Analiza dos numeros ingresados y determina si  
ambos son pares o ambos son impares.  
Entre un número:3  
Entre otro numero7  
Ambos numeros son impares
```

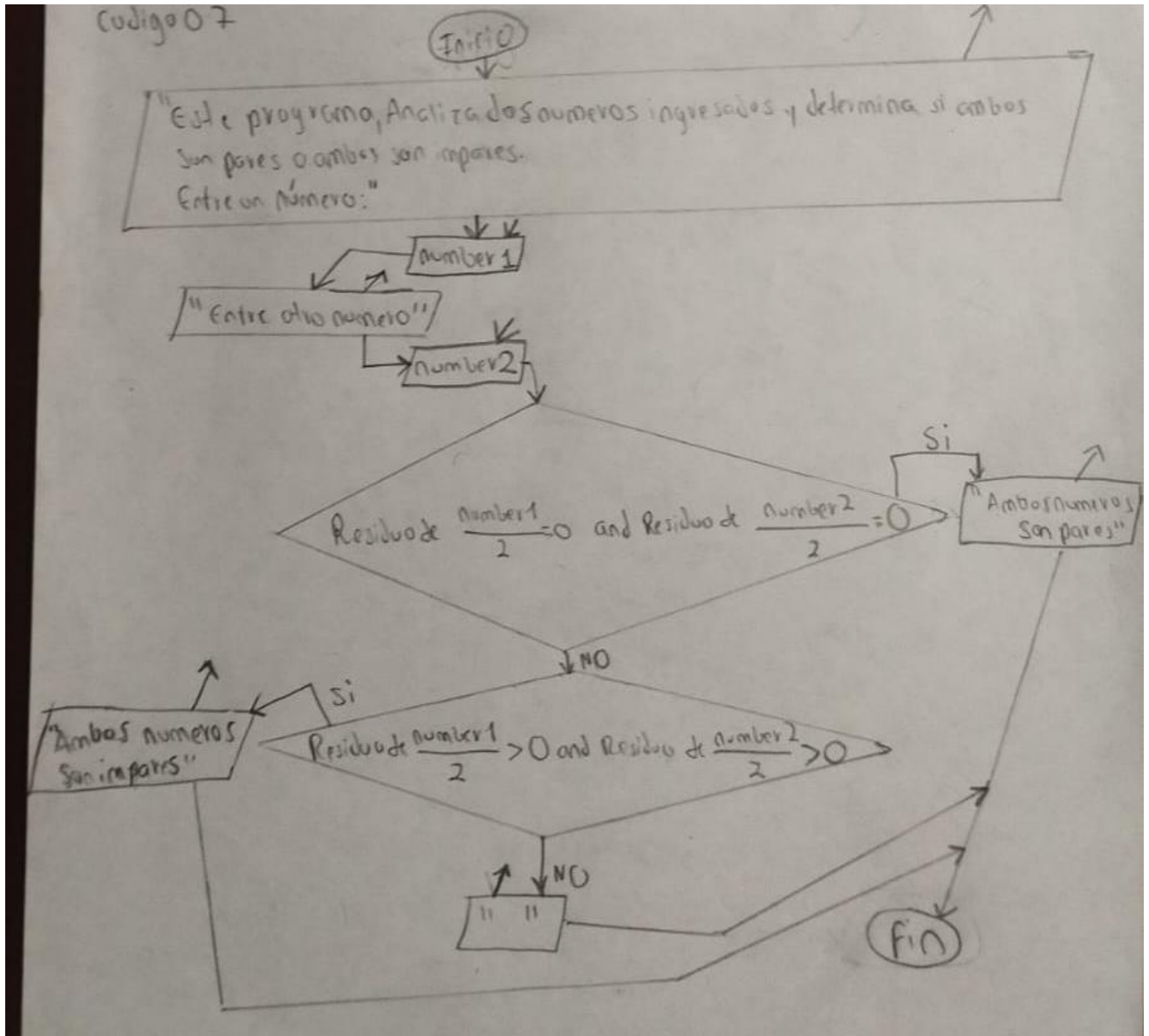
```
Este programa, Analiza dos numeros ingresados y determina si  
ambos son pares o ambos son impares.  
Entre un número:3  
Entre otro numero4  
>
```

Datos entrada y salida:

Input: 2 – 4 (números ingresados por el usuario)

Output: Ambos números son pares

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 22:10

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa analiza dos numeros ingresados por el usuario y analiza si ambos son pares o

ambos son impares

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number1 se utiliza para almacenar el primer numero ingresado por el usuario
number2 se utiliza para almacenar el segundo numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeNumbers)

(printf "Este programa, Analiza dos numeros ingresados y determina si ambos son pares o ambos son impares.

Entre un número:")

(define number1 (read))

(printf "Entre otro numero")

(define number2 (read))

```
( if (and ( = ( remainder number1 2 ) 0 )( = (remainder number2 2 ) 0 ))  
  ( printf "Ambos numeros son pares " )  
  ( if (and ( > ( remainder number1 2 ) 0 )( > (remainder number2 2 ) 0 ))  
    ( printf "Ambos numeros son impares")  
    (printf "")  
  )))  
( AnalyzeNumbers )
```

Código 8: Que tome tres números y diga si la multiplicación de los dos primeros es igual al tercero.

Enlace: <https://youtu.be/UTBST26qyto>

Salida en pantalla:

```
Este programa, Analiza tres numeros ingresados y determina si
la multiplicacion de los 2 primeros es
igual al tercer numero
Entre un número:2
Entre otro numero4
Entre otro numero8
La multiplicacion de 2 y 4 es igual a 8
```

```
Este programa, Analiza tres numeros ingresados y determina si
la multiplicacion de los 2 primeros es
igual al tercer numero
Entre un número:7
Entre otro numero4
Entre otro numero35
```

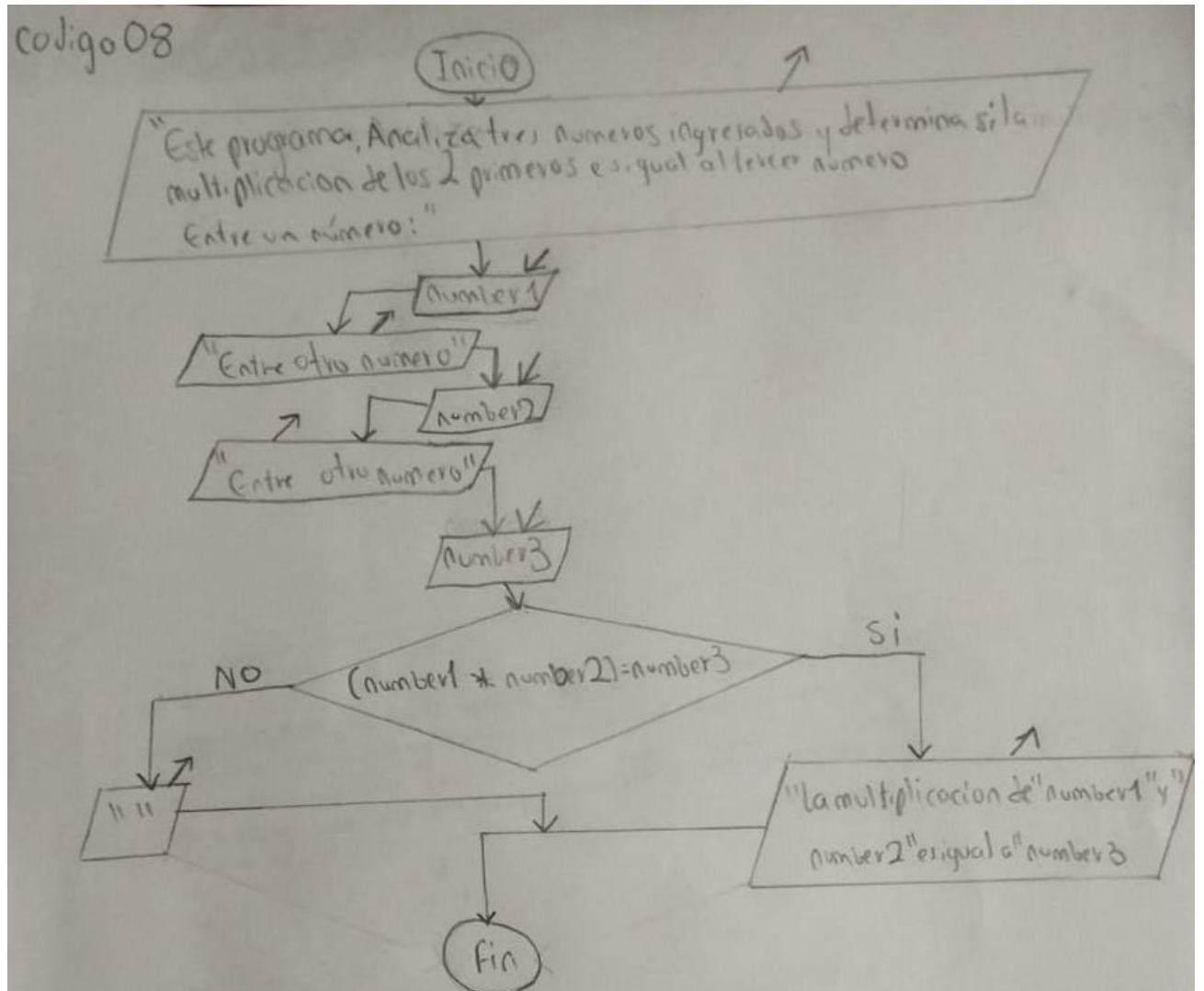
```
Este programa, Analiza tres numeros ingresados y determina si
la multiplicacion de los 2 primeros es
igual al tercer numero
Entre un número:10
Entre otro numero30
Entre otro numero300
La multiplicacion de 10 y 30 es igual a 300
```

Datos entrada y salida:

Input: 2 – 4 – 8 (ingresados por el usuario)

Output: la multiplicación de 2 y 4 es igual a 8

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 22:22

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: El programa analiza tres numeros ingresados por el usuario y analiza si la multiplicaciones

de los 2 primeros es igual al tercer numero

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : number1 se utiliza para almacenar el primer numero ingresado por el usuario

number2 se utiliza para almacenar el segundo numero ingresado por el usuario

number3 se utiliza para almacenar el tercer numero ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeNumbers)

(printf"Este programa, Analiza tres numeros ingresados y determina si la multiplicacion de los 2 primeros es

igual al tercer numero

Entre un número:")

(define number1 (read))


```
( printf "Entre otro numero" )  
( define number2 (read ))  
( printf "Entre otro numero")  
( define number3 (read ))  
  ( if ( = ( * number1 number2 ) number3 )  
    ( printf "La multiplicacion de ~a y ~a es igual a ~a" number1 number2 number3 )  
    ( printf " " )  
  ))  
( AnalyzeNumbers )
```

Código 9: Un hombre desea saber cuanto dinero se genera por concepto de intereses sobre la cantidad que tiene en inversión

en el banco. El decidirá reinvertir los intereses siempre y cuando estos excedan a \$7,000, y en ese caso desea saber

cuanto dinero tendrá finalmente en su cuenta.

Enlace: <https://youtu.be/71CiaXPb3uM>

Salida en pantalla:

```
Este programa lee dos valores. Interés(%) y Capital. Si el
dinero recibido por intereses es
mayor que $7000, se le indicara al inversionista que invierta,
de lo contrario se le indicara
que no debe invertir
Entre el interés en % :10
Entre el capital a invertir en $:100
NO INVIERTA
```

```
Este programa lee dos valores. Interés(%) y Capital. Si el
dinero recibido por intereses es
mayor que $7000, se le indicara al inversionista que invierta,
de lo contrario se le indicara
que no debe invertir
Entre el interés en % :10
Entre el capital a invertir en $:100000
INVIERTA, SU SALDO SERA $110000
```

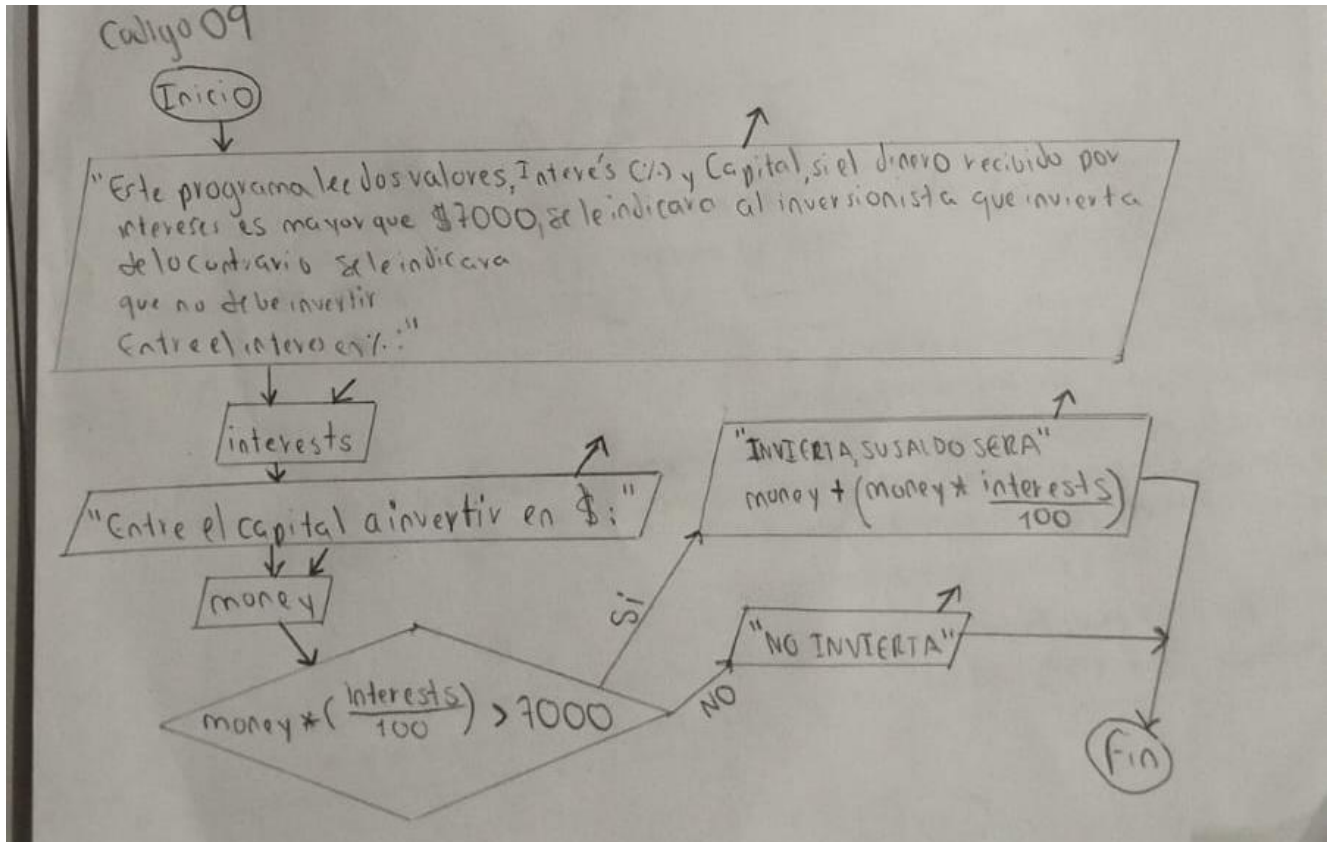
```
Este programa lee dos valores. Interés(%) y Capital. Si el
dinero recibido por intereses es
mayor que $7000, se le indicara al inversionista que invierta,
de lo contrario se le indicara
que no debe invertir
Entre el interés en % :45
Entre el capital a invertir en $:3000
NO INVIERTA
```

Datos entrada y salida:

Input: 10 – 100 (intereses capital)

Output: NO INVIERTA

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 22:41

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa lee dos valores. Interés(%) y Capital. Si el dinero recibido por intereses es

mayor que \$7000, se le indicara al inversionista que invierta, de lo contrario se le indicara que no debe invertir

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : interests se utiliza para almacenar el valor de los intereses ingresados por el usuario
money se utiliza para guardar el valor del capital que ingreso el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeInterests)

(printf"Este programa lee dos valores. Interés(%) y Capital. Si el dinero recibido por intereses es mayor que \$7000, se le indicara al inversionista que invierta, de lo contrario se le indicara que no debe invertir

Entre el interés en % :"

(define interests (read))

```
( printf "Entre el capital a invertir en $:" )  
( define money (read ))  
(if (> ( * money ( / interests 100)) 7000)  
  ( printf "INVIERTA, SU SALDO SERA $~a" ( + money ( * money ( / interests 100)) ))  
  ( printf "NO INVIERTA")  
  
  ))  
( AnalyzeInterests )
```

Código 10: Determinar si un alumno aprueba a reprueba un curso, sabiendo que aprobara si su promedio de tres calificaciones

es mayor o igual a 3,0 y reprueba en caso contrario.

Enlace: <https://youtu.be/4YaYw86tQdA>

Salida en pantalla:

```
Este programa, Analiza tres calificaciones de un alumno y
determina si este aprueba o reprueba el curso
Entre una calificacion:3
Entre otra calificacion3
Entre otra calificaciones3
El alumno aprobo el curso
```

```
Este programa, Analiza tres calificaciones de un alumno
determina si este aprueba o reprueba el curso
Entre una calificacion:5
Entre otra calificacion1
Entre otra calificaciones3
El alumno aprobo el curso
```

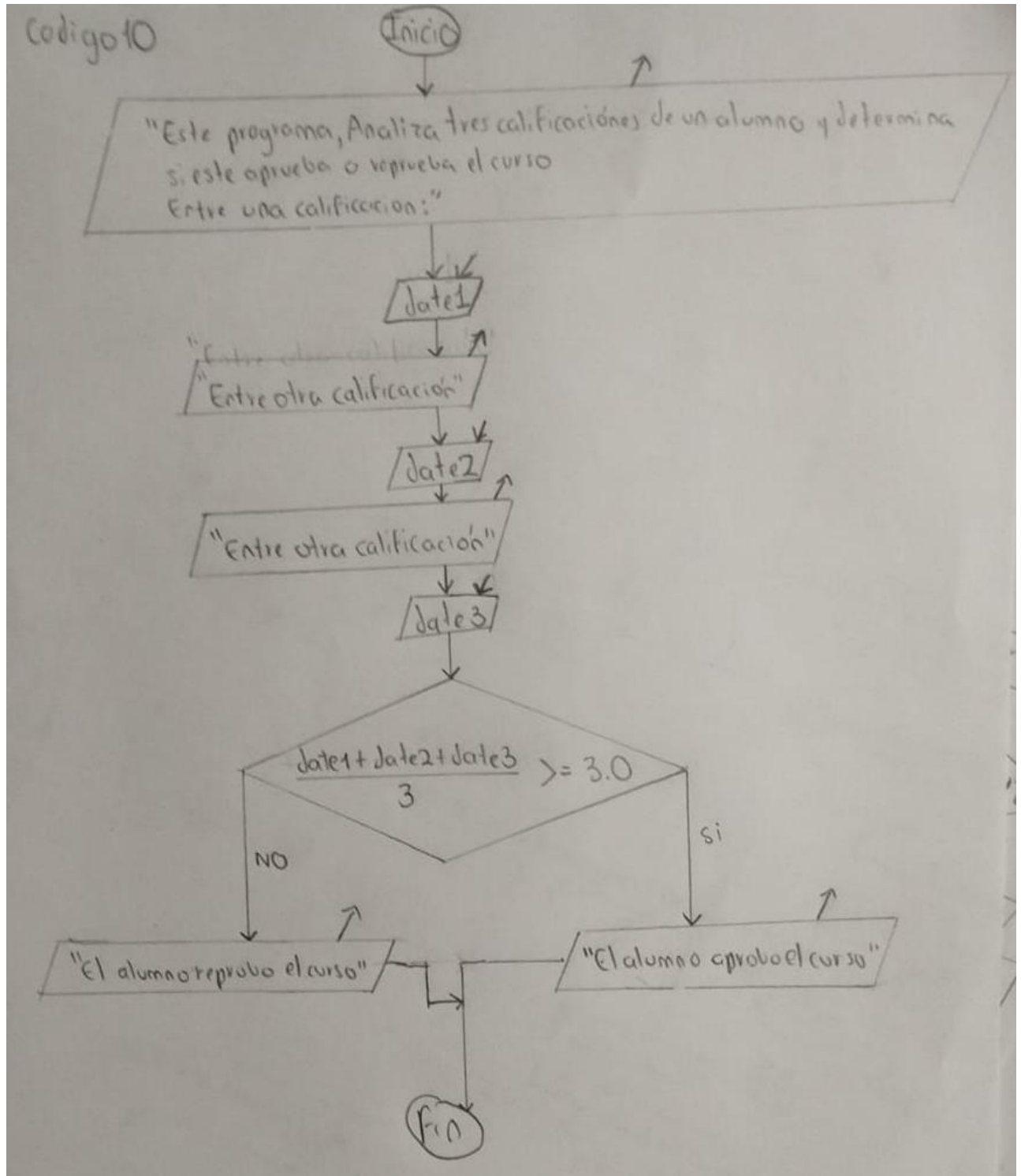
```
Este programa, Analiza tres calificaciones de un alumno y
determina si este aprueba o reprueba el curso
Entre una calificacion:1
Entre otra calificacion2
Entre otra calificaciones3
El alumno reprobo el curso
```

Datos entrada y salida:

Input: 3 – 3 – 3 (calificaciones ingresadas por el usuario)

Output: el alumno aprobó el curso

DFD:



Código:

#lang racket

#|.

-Fecha:01/03/2024

-Hora: 22:52

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa, Analiza tres calificaciones de un alumno y determina si este aprueba o reprueba el curso

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : date1 se utiliza para almacenar el valor de la primera calificacion

date2 se utiliza para almacenar el valor de la segunda calificacion

date3 se utiliza para almacenar el valor de la tercera calificacion

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeDates)

(printf "Este programa, Analiza tres calificaciones de un alumno y determina si este aprueba o reprueba el curso

Entre una calificacion:")

(define date1 (read))

(printf "Entre otra calificacion")

(define date2 (read))


```
( printf "Entre otra calificaciones")  
( define date3 (read ))  
  ( if ( >= ( / ( + date1 date2 date3 ) 3 ) 3.0)  
    ( printf "El alumno aprobo el curso")  
    ( printf "El alumno reprobo el curso " )  
    ))  
( AnalyzeDates )
```

Código 11: Analiza el valor de compra de una persona y aplica el descuento si es necesario

Enlace: <https://youtu.be/YmzBwnZFBCs>

Salida en pantalla:

```
El siguiente programa analiza el valor de su compra y determina  
si se le aplica un descuento  
o no, y devuelve el valor final a pagar  
Ingrese el valor de su compra: 150000  
El valor de su compra es de 120000.0
```

```
El siguiente programa analiza el valor de su compra y determina  
si se le aplica un descuento  
o no, y devuelve el valor final a pagar  
Ingrese el valor de su compra: 100000  
El valor de su compra es de 100000
```

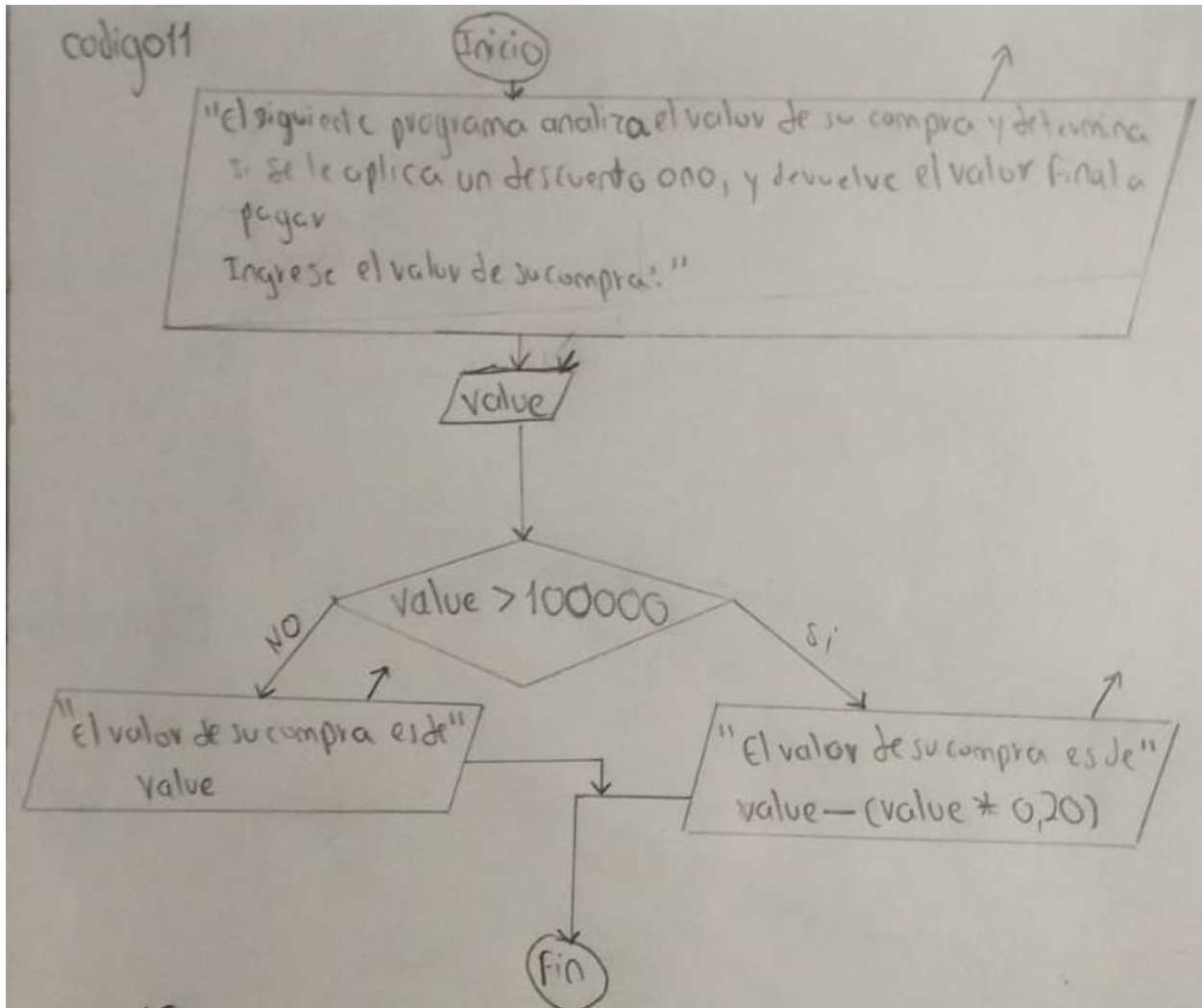
```
El siguiente programa analiza el valor de su compra y determina  
si se le aplica un descuento  
o no, y devuelve el valor final a pagar  
Ingrese el valor de su compra: 100001  
El valor de su compra es de 80000.8
```

Datos entrada y salida:

Input: 150000 (valor de compra)

Output: 120000.0

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 14:19

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa analiza el valor de compra de una persona y aplica el descuento si el valor supera

100000 , imprimiendo el valor denuevo

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : value se utiliza para guardar el valor de la compra ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePrice)

 (printf "El siguiente programa analiza el valor de su compra y determina si se le aplica un descuento

o no, y devuelve el valor final a pagar

Ingrese el valor de su compra: ")

 (define value (read))

 (if (> value 100000)

 (printf " El valor de su compra es de ~a" (- value (* value 0.20)))

```
( printf " El valor de su compra es de ~a" value )  
))  
( AnalyzePrice )
```

Código 12: Este programa analiza el numero de horas trabajadas a la semana , y con respecto a estas devuelve el salario semanal correspondiente

Enlace: <https://youtu.be/4GHLqSkJ02M>

Salida en pantalla:

```
El siguiente programa calcula su salario semanal con respecto a
los siguientes parametros:
-Si trabaja 40 horas o menos se le paga $16 por hora
- Si trabaja mas de 40 horas se le paga $16 por cada una de las
primeras 40 horas y $20 por cada hora extra.
Ingrese su numero de horas trabajadas en la semana:55
su salario semanal es de 940
```

```
El siguiente programa calcula su salario semanal con respecto a
los siguientes parametros:
-Si trabaja 40 horas o menos se le paga $16 por hora
- Si trabaja mas de 40 horas se le paga $16 por cada una de las
primeras 40 horas y $20 por cada hora extra.
Ingrese su numero de horas trabajadas en la semana:34
su salario semanal es de 544
```

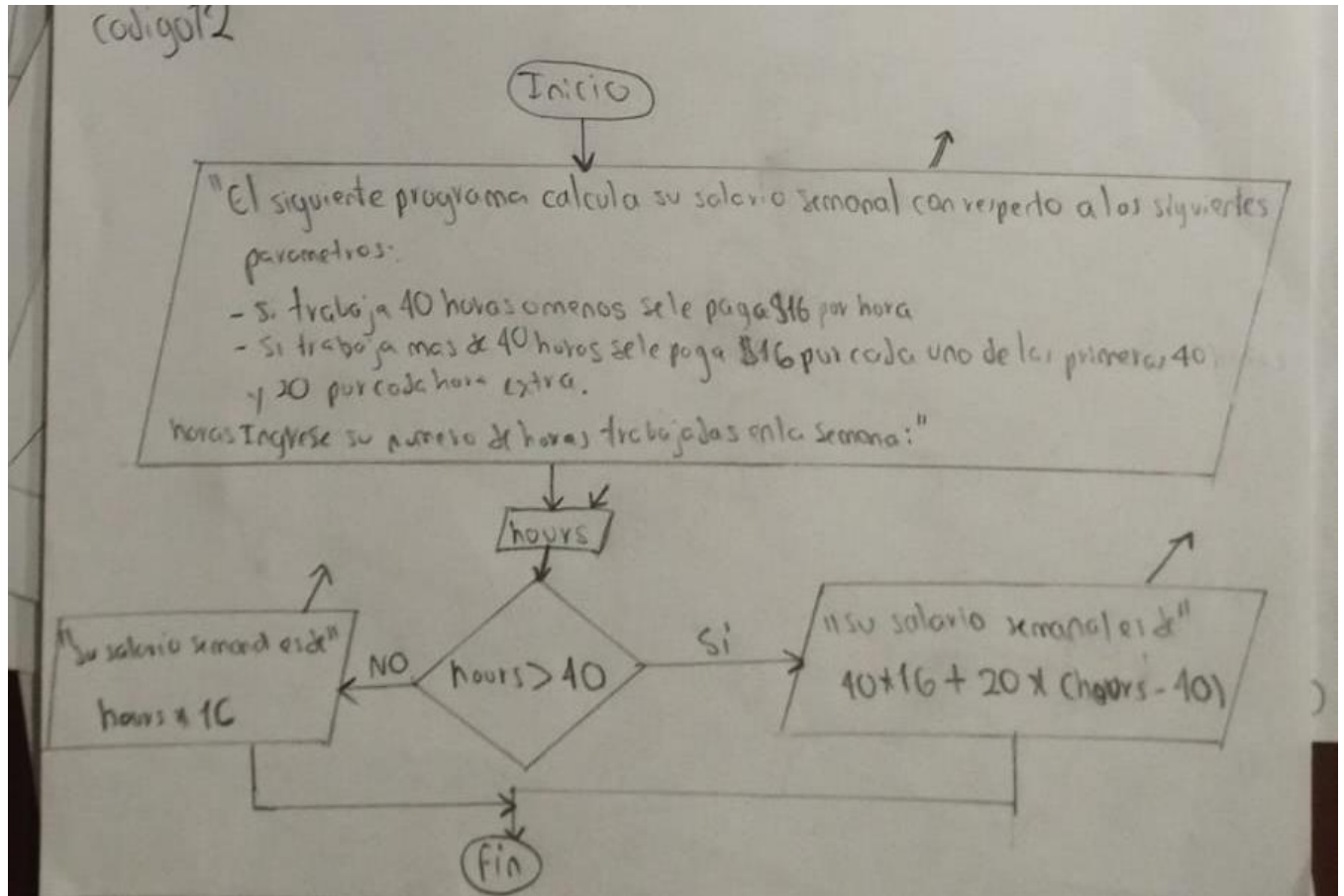
```
El siguiente programa calcula su salario semanal con respecto a
los siguientes parametros:
-Si trabaja 40 horas o menos se le paga $16 por hora
- Si trabaja mas de 40 horas se le paga $16 por cada una de las
primeras 40 horas y $20 por cada hora extra.
Ingrese su numero de horas trabajadas en la semana:41
su salario semanal es de 660
```

Datos entrada y salida:

Input: 55 (números de horas trabajadas)

Output: su salario semanal es de 940

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 14:33

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa analiza el numero de horas trabajadas a la semana , y con respecto a estas

devuelve el salario semanal correspondiente

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos, de lo contrario, se expresará la violación del contrato

-Identificador : hours se utiliz para guardar el numero de horas trabajadas semanalmente que ingresa el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeSalaries)

 (printf "El siguiente programa calcula su salario semanal con respecto a los siguientes
parametros:

-Si trabaja 40 horas o menos se le paga \$16 por hora

- Si trabaja mas de 40 horas se le paga \$16 por cada una de las primeras 40 horas y \$20 por cada hora extra.

Ingrese su numero de horas trabajadas en la semana:")

 (define hours (read))


```
( if ( > hours 40 )  
  ( printf " su salario semanal es de ~a" ( + ( * 40 16 ) ( * 20 ( - hours 40 ) ) ) )  
  ( printf " su salario semanal es de ~a" ( * hours 16 ) )  
  )  
( AnalyzeSalaries )
```

Código 13: Calcula el numero de calorías consumidas por una persona dependiendo la actividad realizada

y el tiempo de duracion

Enlace: https://youtu.be/_ddL5AArfl0

Salida en pantalla:

```
El siguiente programa calcula la cantidad de calorías consumidas en base a la actividad realizada y el tiempo de esta ,con los siguientes parametros:  
- dormir consume 1.08 calorías por minuto.  
- reposo consume 1.66 calorías por minuto.  
SALVEDADES: No se garantizan resultados si al ingresar la actividad no se pone extricatamente una de estas dos actividades ENTRE COMILLAS sin variaciones en la escritura dormir - reposo .  
Ingrese la actividad que realizo"dormir"  
Ingrese el numero de minutos que hizo la actividad 16  
El numero de calorías que ha consumido por dormir 16 minutos son 17.28
```

```
El siguiente programa calcula la cantidad de calorías consumidas en base a la actividad realizada y el tiempo de esta ,con los siguientes parametros:  
- dormir consume 1.08 calorías por minuto.  
- reposo consume 1.66 calorías por minuto.  
SALVEDADES: No se garantizan resultados si al ingresar la actividad no se pone extricatamente una de estas dos actividades ENTRE COMILLAS sin variaciones en la escritura dormir - reposo .  
Ingrese la actividad que realizo"reposo"  
Ingrese el numero de minutos que hizo la actividad 67  
El numero de calorías que ha consumido por estar sentado en reposo 67 minutos son 111.22
```

```
El siguiente programa calcula la cantidad de calorías consumidas en base a la actividad realizada y el tiempo de esta ,con los siguientes parametros:  
- dormir consume 1.08 calorías por minuto.  
- reposo consume 1.66 calorías por minuto.  
SALVEDADES: No se garantizan resultados si al ingresar la actividad no se pone extricatamente una de estas dos actividades ENTRE COMILLAS sin variaciones en la escritura dormir - reposo .  
Ingrese la actividad que realizo"dormir"  
Ingrese el numero de minutos que hizo la actividad 1  
El numero de calorías que ha consumido por dormir 1 minutos son 1.08
```

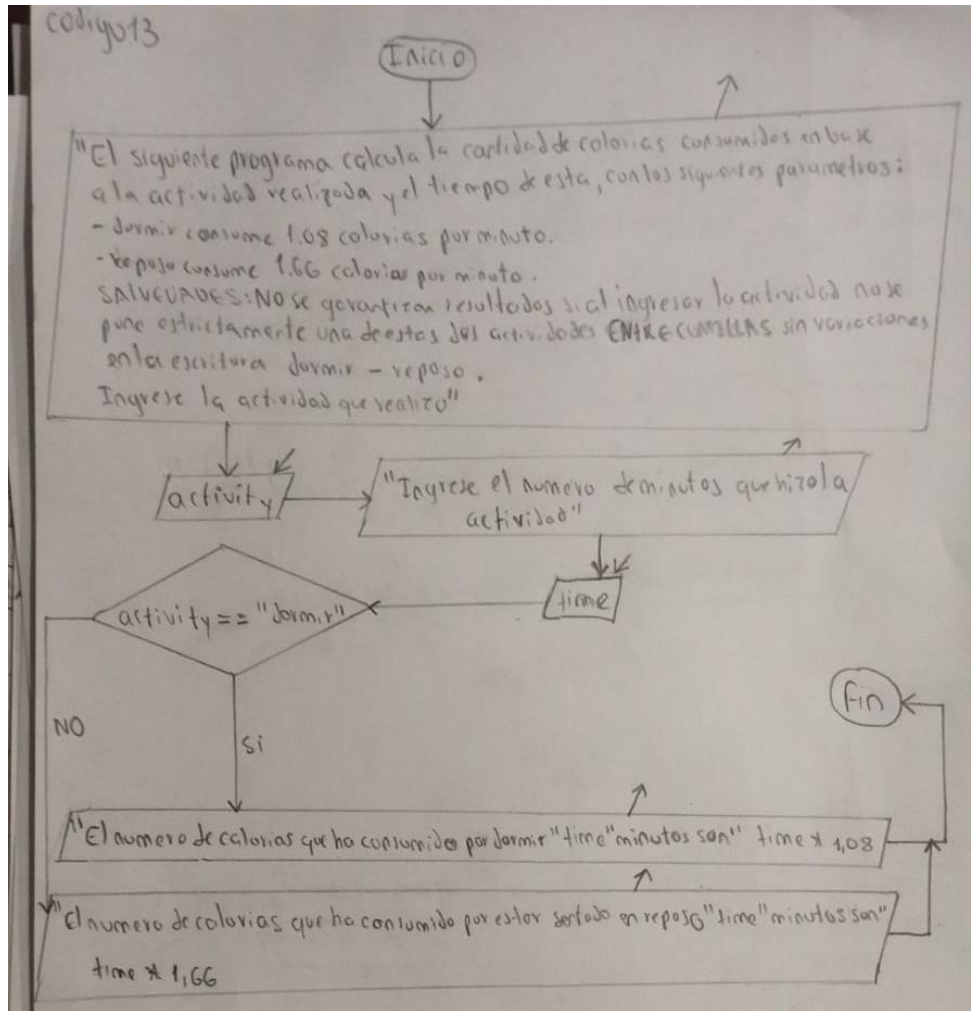
Datos entrada y salida:

Input “dormir” (string ingresado por el usuario)

16 (tiempo en minutos ingresados por el usuario)

Output: El numero de calorías que ha consumido por dormir 16 minutos son 17.28

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 15:13

-Versión de su código: 9

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa calcula el numero de calorías consumidas por una persona dependiendo la actividad realizada

y el tiempo de duracion

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos en el apartado de "Ingrese el numero de minutos que hizo la actividad

", de lo contrario, se expresará la violación del contrato

No se garantizan resultados si al ingresar la actividad no se pone extricatamente una de estas dos actividades ENTRE COMILLAS sin variaciones en la escritura dormir - reposo .

-Identificador : activity se usa para almacenar la actividad que hizo el usuario

time se utiliza para almacenar la cantidad de minutos que se realizo la actividad

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeCalories)

(printf "El siguiente programa calcula la cantidad de calorías consumidas en base a la actividad realizada

y el tiempo de esta ,con los siguientes parametros:

- dormir consume 1.08 calorías por minuto.

- reposo consume 1.66 calorías por minuto.

SALVEDADES: No se garantizan resultados si al ingresar la actividad no se pone extricatamente una de estas dos

actividades ENTRE COMILLAS sin variaciones en la escritura dormir - reposo .

Ingrese la actividad que realizo")

```
( define activity ( read ) )
```

```
( printf "Ingrese el numero de minutos que hizo la actividad " )
```

```
( define time ( read ) )
```

```
( if ( equal? "dormir" activity )
```

```
  ( printf " El numero de calorías que ha consumido por dormir ~a minutos son ~a " time ( * time 1.08 ) )
```

```
  ( printf " El numero de calorías que ha consumido por estar sentado en reposo ~a minutos son ~a " time ( * time 1.66 )
```

```
  )))
```

```
( AnalyzeCalories )
```

Código 14: Que imprima de un artículo, clave, precio original y su precio con descuento. El descuento lo hace en base a la

clave, si la clave es 1 el descuento es del 10% y si la clave es 2 el descuento en del 20% (solo existen dos claves).

Datos de entrada: Clave, PrecioOriginal

Enlace: <https://youtu.be/1XyRSixy4NQ>

Salida en pantalla:

```
Este programa calcula y imprime el precio con
descuento de un artículo a la vez que imprime la
clave y
precio original del mismo. El precio se calcula con
los siguientes parametros:
si la clave es 1 el descuento es del 10% y si la
clave es 2 el descuento en del 20% (solo existen dos
claves).
SALVEDAD: No se garantizan resultados cuando ingresen
valores para la clave fuera del rango de 1 y 2 .
  Ingrese el valor del artículo : 1500
Ingrese la clave (1 o 2) 2
El precio original del artículo es de : 1500
La clave del artículo es : 2
El precio con descuento del artículo es de : 1200.0 >
```

```
Este programa calcula y imprime el precio con
descuento de un artículo a la vez que imprime la
clave y
precio original del mismo. El precio se calcula con
los siguientes parametros:
si la clave es 1 el descuento es del 10% y si la
clave es 2 el descuento en del 20% (solo existen dos
claves).
SALVEDAD: No se garantizan resultados cuando ingresen
valores para la clave fuera del rango de 1 y 2 .
  Ingrese el valor del artículo : 2000
Ingrese la clave (1 o 2) 1
El precio original del artículo es de : 2000
La clave del artículo es : 1
El precio con descuento del artículo es de : 1800.0 > |
```

```
Este programa calcula y imprime el precio con
descuento de un articulo a la vez que imprime la
clave y
precio original del mismo. El precio se calcula con
los siguientes parametros:
si la clave es 1 el descuento es del 10% y si la
clave es 2 el descuento en del 20% (solo existen dos
claves).
SALVEDAD: No se garantizan resultados cuando ingresen
valores para la clave fuera del rango de 1 y 2 .
Ingrese el valor del articulo : 4900
Ingrese la clave (1 o 2) 2
El precio original del articulo es de : 4900
La clave del articulo es : 2
El precio con descuento del articulo es de : 3920.0 >
```

Datos entrada y salida:

Input: 1500 (precio original del articulo)

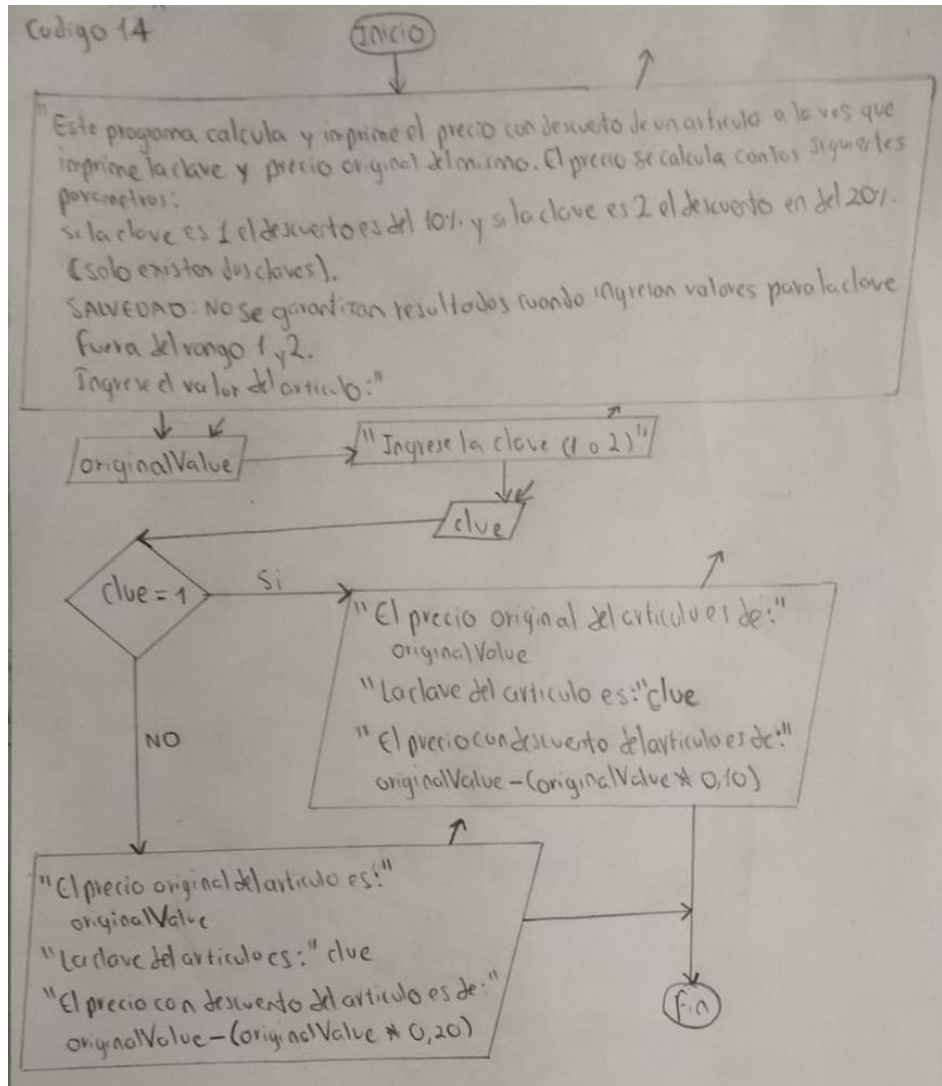
2 (clave del articulo)

Output: El precio original del articulo es de : 1

La clave del articulo es : 2

El precio con descuento del articulo es de : 0.8

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 15:45

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa imprime el precio original , clave y precio con descuento de un articulo

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato

-No se garantizan resultados cuando ingresen valores para la clave fuera del rango de 1 y 2 .

-Identificador : originalValue se utiliza para almacenar el precio original del articulo

clue se utiliza para almacenar la clave del articulo

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePrice)

(printf "Este programa calcula y imprime el precio con descuento de un articulo a la ves que imprime la clave y

precio original del mismo. El precio se calcula con los siguientes parametros:

si la clave es 1 el descuento es del 10% y si la clave es 2 el descuento en del 20% (solo existen dos claves).

SALVEDAD: No se garantizan resultados cuando ingresen valores para la clave fuera del rango de 1 y 2 .

Ingrese el valor del articulo : ")

```
( define originalValue ( read ) )
```

```
( printf "Ingrese la clave (1 o 2) "
```

```
( define clue ( read ) )
```

```
( if ( = clue 1 )
```

```
    ( printf "El precio original del articulo es de : ~a
```

La clave del articulo es : ~a

El precio con descuento del articulo es de : ~a " originalValue clue (- originalValue (* originalValue 0.10)))

```
    ( printf "El precio original del articulo es de : ~a
```

La clave del articulo es : ~a

El precio con descuento del articulo es de : ~a " originalValue clue (- originalValue (* originalValue 0.20)))

```
    ))
```

```
( AnalyzePrice )
```

Código 15: calcula el precio total a pagar por la compra de camisetas en funcion de la cantidad y precio por camiseta

Enlace: <https://youtu.be/urMamiUwpxU>

Salida en pantalla:

```
Este programa calcula el total a pagar por la compra
de camisetas teniendo en cuenta los siguientes
parametros:
Si se compran tres camisas o mas se aplica un
descuento del 20% sobre el total de la compra y si
son menos de tres camisas
    un descuento del 10%.
Ingrese el numero de casitas compradas1
Ingrese el precio por camiseta100
El total a pagar por la compra de las camisetas es de
90.0
```

```
Este programa calcula el total a pagar por la compra
de camisetas teniendo en cuenta los siguientes
parametros:
Si se compran tres camisas o mas se aplica un
descuento del 20% sobre el total de la compra y si
son menos de tres camisas
    un descuento del 10%.
Ingrese el numero de casitas compradas10
Ingrese el precio por camiseta3000
El total a pagar por la compra de las camisetas es de
24000.0
```

```
Este programa calcula el total a pagar por la compra
de camisetas teniendo en cuenta los siguientes
parametros:
Si se compran tres camisas o mas se aplica un
descuento del 20% sobre el total de la compra y si
son menos de tres camisas
    un descuento del 10%.
Ingrese el numero de casitas compradas7
Ingrese el precio por camiseta35672
El total a pagar por la compra de las camisetas es de
199763.2
```

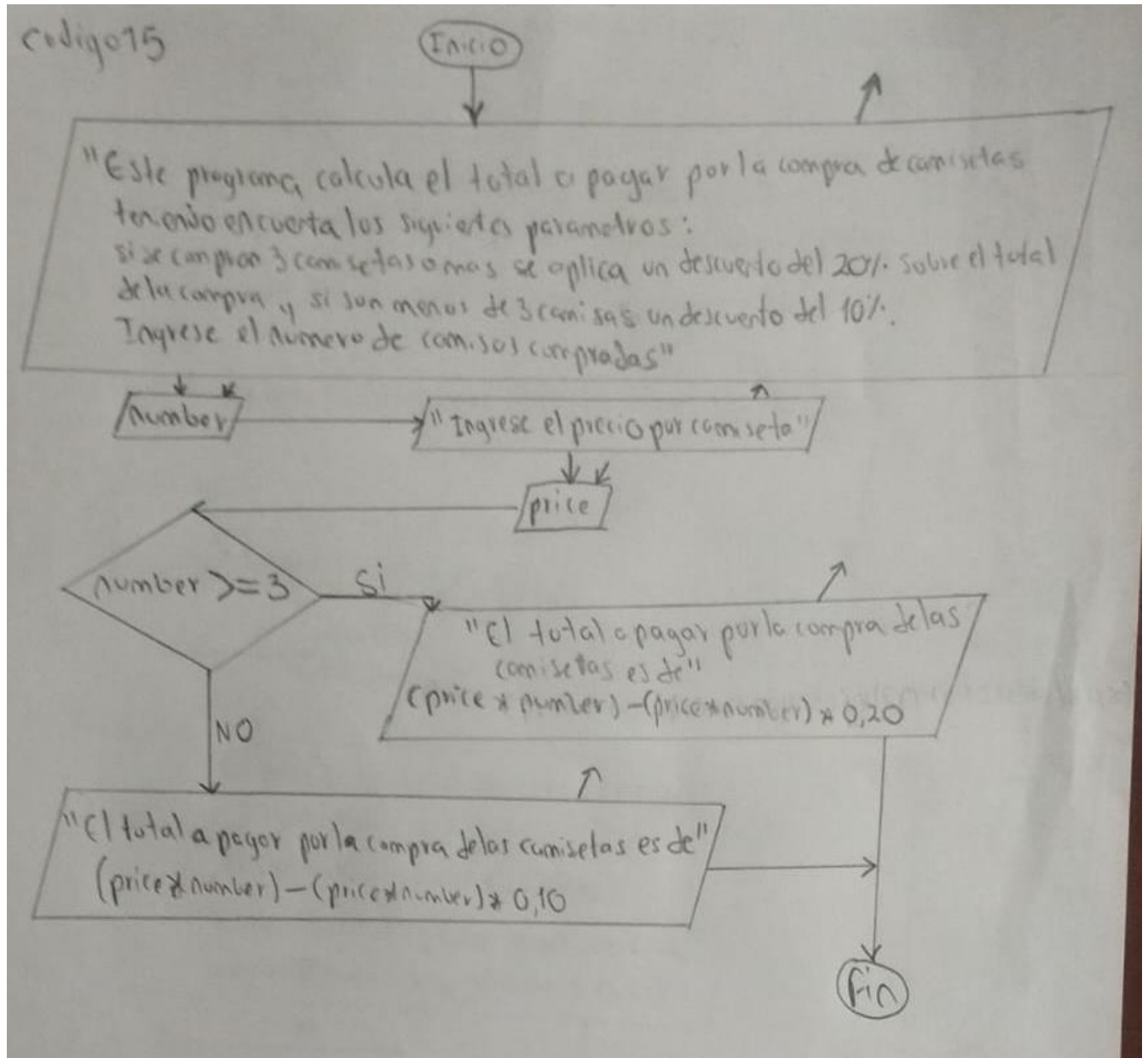
Datos entrada y salida:

Input: 1 (número de camisetas compradas)

100 (precio por camiseta)

Output: El total a pagar por la compra de las camisetas es de 90.0

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 16:01

-Versión de su código: 1

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa calcula el precio total a pagar por la compra de camisetas en funcion de la cantidad y precio

por camiseta

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-Identificador : number se utiliza para almacenar el numero de camisetas

price se utiliza para almacenar el precio por camiseta

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePrice)

(printf "Este programa calcula el total a pagar por la compra de camisetas teniendo en cuenta los siguientes parametros:

Si se compran tres camisas o mas se aplica un descuento del 20% sobre el total de la compra y si son menos de tres camisas

un descuento del 10%.

Ingrese el numero de camisas compradas")

(define number (read))

```
( printf "Ingrese el precio por camiseta")  
( define price ( read ) )  
( if ( >= number 3 )  
  ( printf "El total a pagar por la compra de las camisetas es de ~a"  
    ( - ( * price number ) ( * ( * price number ) 0.20 ) ) )  
  ( printf "El total a pagar por la compra de las camisetas es de ~a"  
    ( - ( * price number ) ( * ( * price number ) 0.10 ) ) )  
  ) )  
( AnalyzePrice )
```

Código 16: calcula cuanto es la cantidad que la empresa tiene que invertir , tomar de un prestamo del banco , sacar con credito del fabricante y el interes cobrado por el fabricante en base al monto total de la compra.

Enlace: <https://youtu.be/h3SnbMdM-o8>

Salida en pantalla:

```
Este programa calcula cuanto es la cantidad que la
empresa tiene que invertir , tomar de un prestamo del
banco , sacar con credito del fabricante y el interes
cobrado por el fabricante
en base al monto total de la compra.
Ingrese el valor de costo por pieza :100
Ingrese el numero de piezas :600
La cantidad que se debe invertir es de 42000.0
El valor del prestamo del banco es 0
El valor a credito con el fabricante es de 18000.0
El valor del interes del credito del fabricantes es
de 3600.0
```

```
Este programa calcula cuanto es la cantidad que la
empresa tiene que invertir , tomar de un prestamo del
banco , sacar con credito del fabricante y el interes
cobrado por el fabricante
en base al monto total de la compra.
Ingrese el valor de costo por pieza :100000
Ingrese el numero de piezas :25
La cantidad que se debe invertir es de 1375000.0
El valor del prestamo del banco es 750000.0
El valor a credito con el fabricante es de 375000.0
El valor del interes del credito del fabricantes es
de 75000.0
```

```
Este programa calcula cuanto es la cantidad que la
empresa tiene que invertir , tomar de un prestamo de
banco , sacar con credito del fabricante y el intere
cobrado por el fabricante
en base al monto total de la compra.
Ingrese el valor de costo por pieza :50000
Ingrese el numero de piezas :10
La cantidad que se debe invertir es de 350000.0
El valor del prestamo del banco es 0
El valor a credito con el fabricante es de 150000.0
El valor del interes del credito del fabricantes es
de 30000.0
```


Datos entrada y salida:

Input: 100 (CostoPieza)

600 (NumeroDePiezas)

Output:

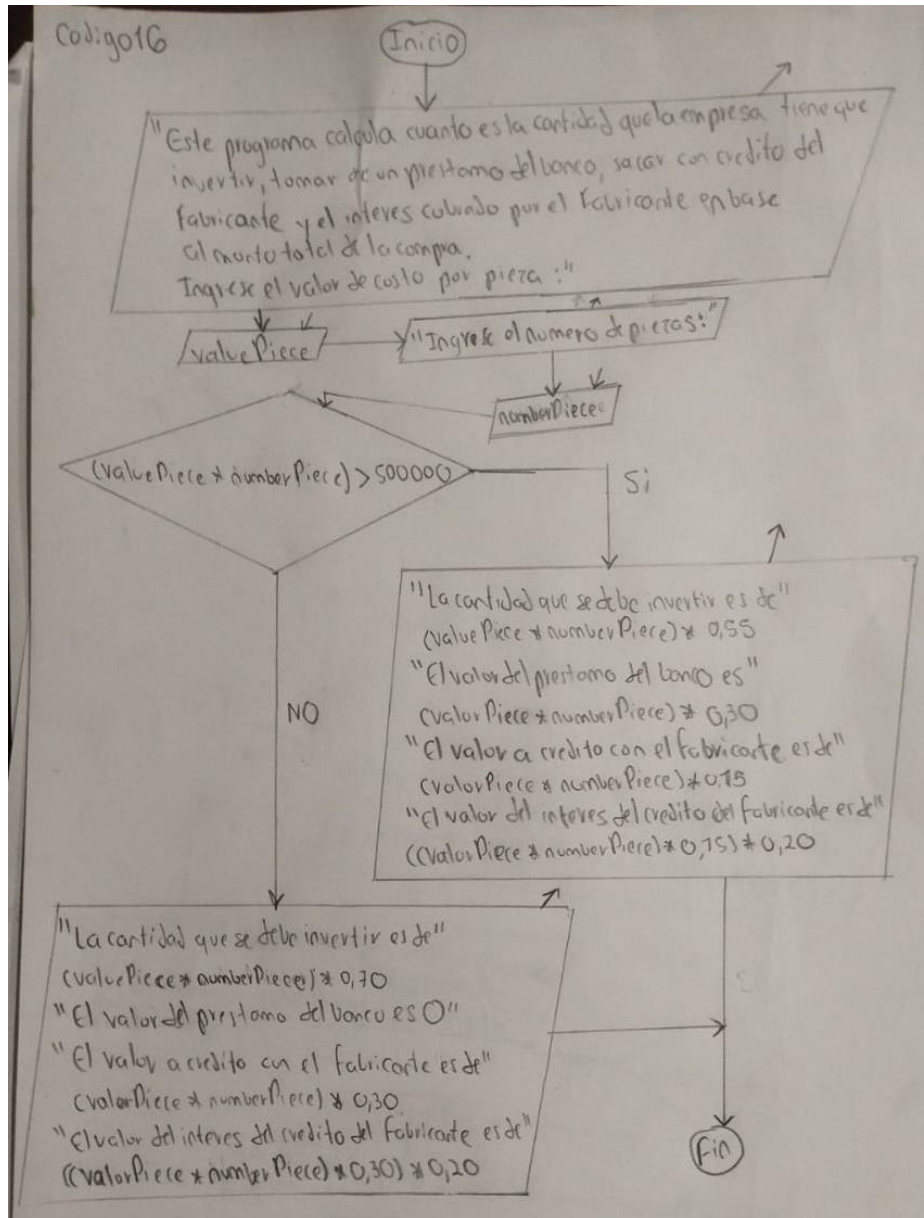
La cantidad que se debe invertir es de 42000.0 (CantidadInvertida)

El valor del prestamo del banco es 0 (ValorPrestamoBanco)

El valor a credito con el fabricante es de 18000.0(ValorCreditoFabricante)

**El valor del interes del credito del fabricantes es de
3600.0(InteresCobradoPorFabricante)**

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 16:37

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa calcula cuanto es la cantidad que la empresa tiene que invertir , tomar de un prestamo del banco , sacar con credito del fabricante y el interes cobrado por el fabricante en base al monto total de la compra.

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-Identificador : valuePiece se utiliza para guardar el valor por pieza

numberPiece se utiliza para guardar el numero de piezas a comprar

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeBuy)

(printf "Este programa calcula cuanto es la cantidad que la empresa tiene que invertir , tomar de un prestamo del banco , sacar con credito del fabricante y el interes cobrado por el fabricante en base al monto total de la compra.

Ingrese el valor de costo por pieza :")

(define valuePiece (read))

(printf "Ingrese el numero de piezas :")

(define numberPiece (read))

```

( if ( > ( * valuePiece numberPiece) 500000 )
  (begin
    ( printf "La cantidad que se debe invertir es de ~a~n" ( * ( * valuePiece numberPiece ) 0.55 ) )
    ( printf "El valor del prestamo del banco es ~a~n" ( * ( * valuePiece numberPiece ) 0.30 ) )
    ( printf "El valor a credito con el fabricante es de ~a~n" ( * ( * valuePiece numberPiece ) 0.15
  ) )

  ( printf "El valor del interes del credito del fabricantes es de ~a~n" ( * ( * ( * valuePiece
numberPiece ) 0.15 ) 0.20 ) ) )

  (begin
    ( printf "La cantidad que se debe invertir es de ~a~n" ( * ( * valuePiece numberPiece ) 0.70 ) )
    ( printf "El valor del prestamo del banco es 0~n" )
    ( printf "El valor a credito con el fabricante es de ~a~n" ( * ( * valuePiece numberPiece ) 0.30
  ) )

    ( printf "El valor del interes del credito del fabricantes es de ~a~n" ( * ( * ( * valuePiece
numberPiece ) 0.30 ) 0.20 ) )

  )

  ))

( AnalyzeBuy )

```

Código 17: Construir un programa que calcule el índice de masa corporal de una persona, donde: $(IMC = \text{peso [kg]} / \text{altura}^2[\text{m}])$ e indique el estado en el que se encuentra esa persona en función del valor de IMC:

Enlace: <https://youtu.be/ygttdkF-hg8>

Salida en pantalla:

```
Este programa calcula el indice de masa corporal en
base a la siguiente ecuacion:
(IMC = peso [kg] /altura2[m])
y indica el estado en que se encuentra la persona.
Ingrese su peso en kg :60
Ingrese su altura en m :1.74
peso normal (saludable)
```

```
Este programa calcula el indice de masa corporal en
base a la siguiente ecuacion:
(IMC = peso [kg] /altura2[m])
y indica el estado en que se encuentra la persona.
Ingrese su peso en kg :90
Ingrese su altura en m :1.65
sobre peso cronico (obesidad de grado II)
```

```
Este programa calcula el indice de masa corporal en
base a la siguiente ecuacion:
(IMC = peso [kg] /altura2[m])
y indica el estado en que se encuentra la persona.
Ingrese su peso en kg :200
Ingrese su altura en m :2.13
obesidad morbida (obesidad de grado IV)
```

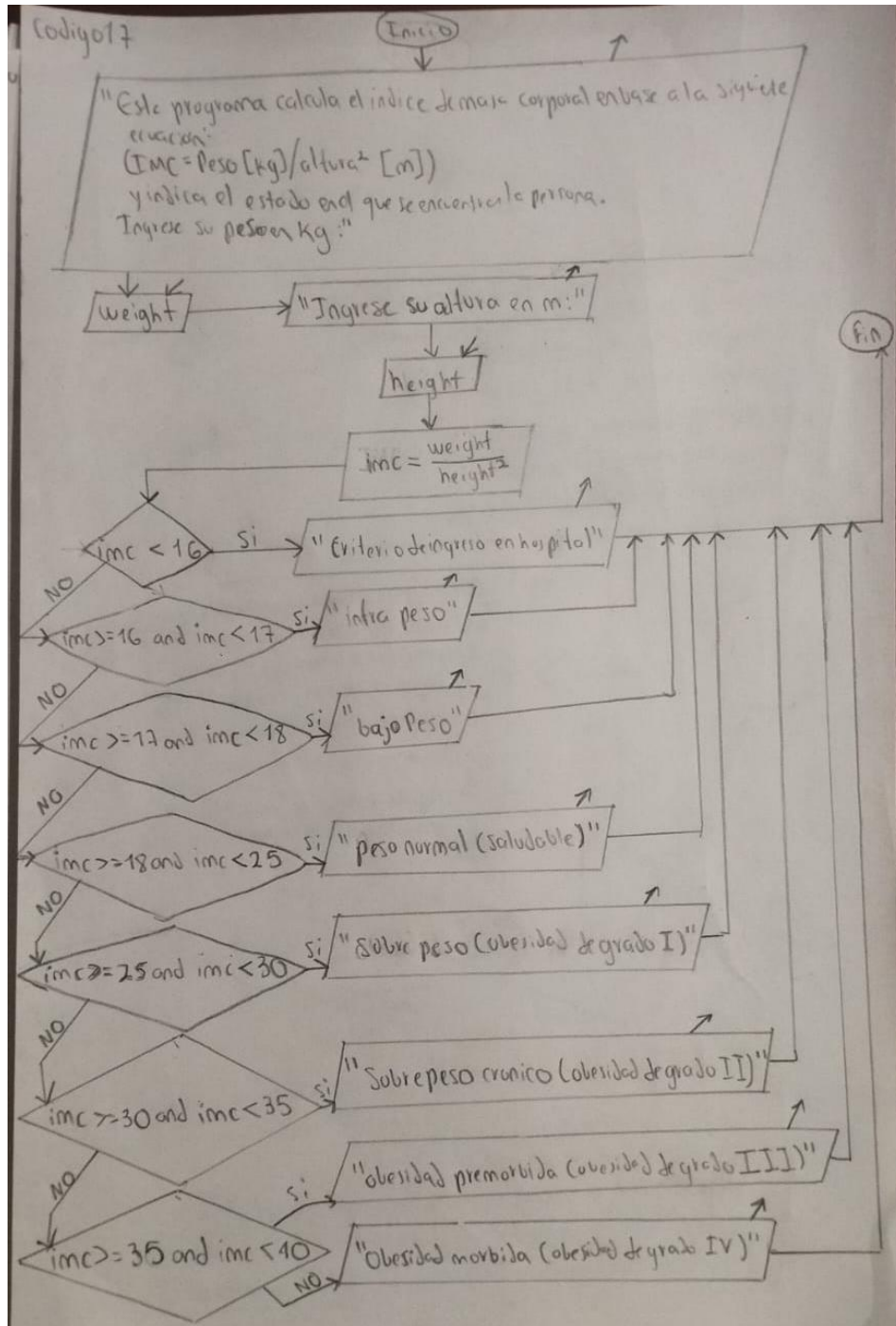
Datos entrada y salida:

Input: 60 (peso en kg)

1.74 (altura en m)

Output: peso normal (saludable)

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 17:11

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa calcula el imc de una persona e indica el estado en el que se encuentra

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-Identificador : weight se utiliza para almacenar el peso en kg ingresado por el usuario

height se utiliza para almacenar la altura en m ingresado por el usuario

imc se utiliza para guardar el valor del indice de masa corporal de los datos ingresados

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeBodyMass)

(printf "Este programa calcula el indice de masa corporal en base a la siguiente ecuacion:

(IMC = peso [kg] /altura2[m])

y indica el estado en que se encuentra la persona.

Ingrese su peso en kg :")

(define weight (read))

(printf "Ingrese su altura en m :")

```

( define height ( read ) )
( define imc ( / weight ( sqr height )))
( if (< imc 16 )
  ( printf "Criterio de ingreso en hospital" )
  ( if ( and ( >= imc 16 ) ( < imc 17 ) )
    ( printf "infra peso" )
    ( if ( and ( >= imc 17 ) ( < imc 18 ) )
      ( printf "bajo peso" )
      ( if ( and ( >= imc 18 ) ( < imc 25 ) )
        ( printf "peso normal (saludable) " )
        ( if ( and ( >= imc 25 ) ( < imc 30 ) )
          ( printf "sobre peso (obesidad de grado I)" )
          ( if ( and ( >= imc 30 ) ( < imc 35 ) )
            ( printf "sobre peso cronico (obesidad de grado II)" )
            ( if ( and ( >= imc 35 ) ( < imc 40 ) )
              ( printf "obesidad premorbida (obesidad de grado III)" )
              ( printf "obesidad morbida (obesidad de grado IV)" )
            )
          )
        )
      )
    )
  )
)
( AnalyzeBodyMass )

```


Código 18: Este programa calcula cuanto es el valor a pagar en funcion del tiempo de estacionamiento en horas y minutos

Enlace: <https://youtu.be/eD6Pv19zDB8>

Salida en pantalla:

```
Este programa calcula cuanto es el valor a pagar en funcion del tiempo de estacionamiento en horas y minutos
teniendo en cuenta que se cobra 1,800 por hora o fraccion.
Ingrese la cantidad de horas :1
Ingrese la cantidad de minutos :60
El valor a pagar por el estacionamiento con un tiempo de 1 horas y 60 minutos es de 3600
```

```
Este programa calcula cuanto es el valor a pagar en funcion del tiempo de estacionamiento en horas y minutos
teniendo en cuenta que se cobra 1,800 por hora o fraccion.
Ingrese la cantidad de horas :1
Ingrese la cantidad de minutos :61
El valor a pagar por el estacionamiento con un tiempo de 1 horas y 61 minutos es de 5400
```

```
Este programa calcula cuanto es el valor a pagar en funcion del tiempo de estacionamiento en horas y minutos
teniendo en cuenta que se cobra 1,800 por hora o fraccion.
Ingrese la cantidad de horas :20
Ingrese la cantidad de minutos :42
El valor a pagar por el estacionamiento con un tiempo de 20 horas y 42 minutos es de 37800
```

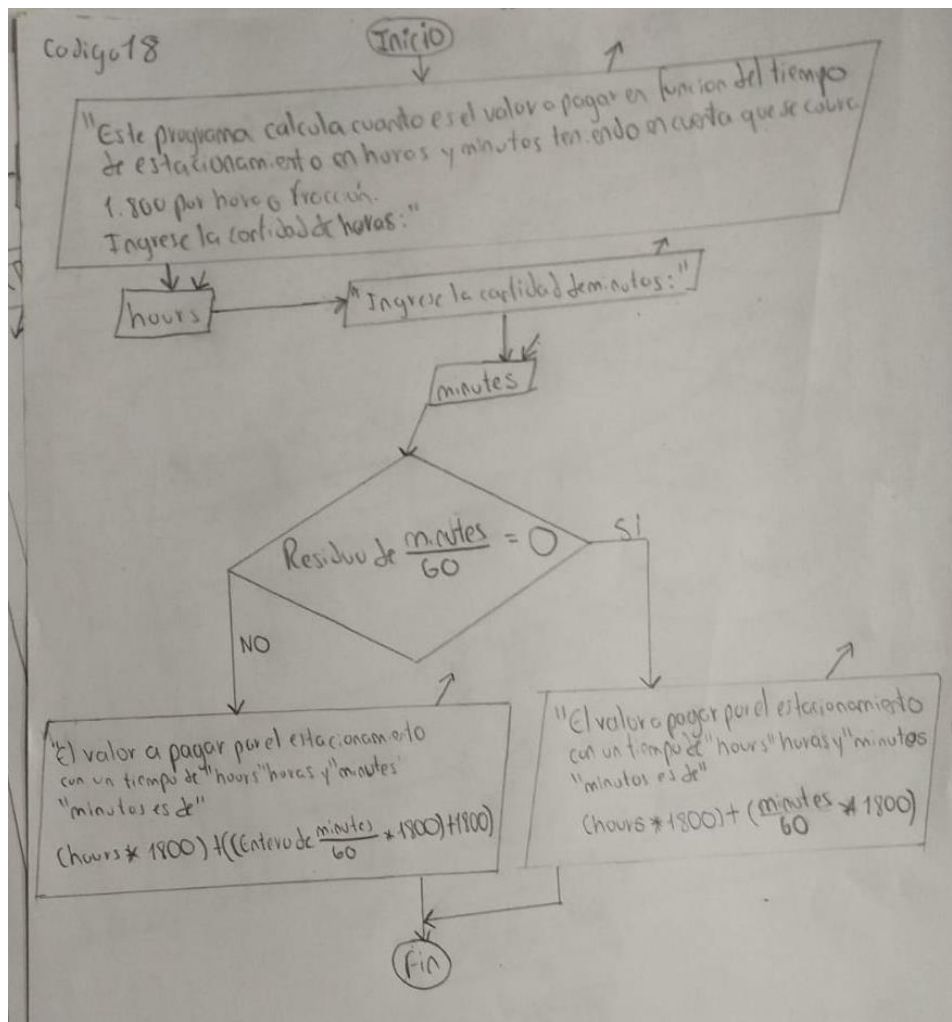
Datos entrada y salida:

Input: 1 (valor de las horas)

60 (valor de los minutos)

Output: El valor a pagar por el estacionamiento con un tiempo de 1 horas y 60 minutos es de 3600

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 18:03

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa calcula cuanto es el valor a pagar en funcion del tiempo de estacionamiento en horas y minutos

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-Identificador : hours se utiliza para almacenar el valor de las horas
minutes se utiliza para almacenar el valor de los minutos

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePrice)

(printf " Este programa calcula cuanto es el valor a pagar en funcion del tiempo de estacionamiento en horas y minutos

teniendo en cuenta que se cobra 1,800 por hora o fraccion.

Ingrese la cantidad de horas :")

(define hours (read))

(printf "Ingrese la cantidad de minutos :")

(define minutes (read))

(if (= (remainder minutes 60) 0)

```
( printf "El valor a pagar por el estacionamiento con un tiempo de ~a horas y ~a minutos es de  
~a"
```

```
hours minutes ( + ( * hours 1800 ) ( * ( / minutes 60 ) 1800 )))
```

```
( printf "El valor a pagar por el estacionamiento con un tiempo de ~a horas y ~a minutos es de  
~a"
```

```
hours minutes ( + ( * hours 1800 ) ( + ( * ( quotient minutes 60 ) 1800 ) 1800 ) ) )
```

```
) )
```

```
( AnalyzePrice )
```

Código 19: Este programa calcula cual es el monto de la compra , el monto del descuento , el monto a pagar y el numero de unidades de obsequio por la compra de cierto producto

Enlace: <https://youtu.be/0kxKBwCGL4o>

Salida en pantalla:

```
Este programa calcula cual es el monto de la compra , el monto del descuento , el monto a pagar y el numero
de unidades de obsequio por la compra de cierto producto teniendo en cuenta que:
un descuento del 15% por la compra de más de 3 docenas y 10% en caso contrario.
Ingrese el precio del producto :100
Ingrese la cantidad que va a comprar :2000
El monto de la compra es de 200000
El monto del descuento es de 30000.0
El monto a pagar es de 170000.0
El numero de unidades de obsequio es de 163

Este programa calcula cual es el monto de la compra , el monto del descuento , el monto a pagar y el numero
de unidades de obsequio por la compra de cierto producto teniendo en cuenta que:
un descuento del 15% por la compra de más de 3 docenas y 10% en caso contrario.
Ingrese el precio del producto :10000
Ingrese la cantidad que va a comprar :15
El monto de la compra es de 150000
El monto del descuento es de 15000.0
El monto a pagar es de 135000.0
El numero de unidades de obsequio es de 0

Este programa calcula cual es el monto de la compra , el monto del descuento , el monto a pagar y el numero
de unidades de obsequio por la compra de cierto producto teniendo en cuenta que:
un descuento del 15% por la compra de más de 3 docenas y 10% en caso contrario.
Ingrese el precio del producto :100
Ingrese la cantidad que va a comprar :45
El monto de la compra es de 4500
El monto del descuento es de 675.0
El monto a pagar es de 3825.0
El numero de unidades de obsequio es de 0
```

Datos entrada y salida:

Input: 100 (precio del producto)

2000 (cantidad del producto)

Output:

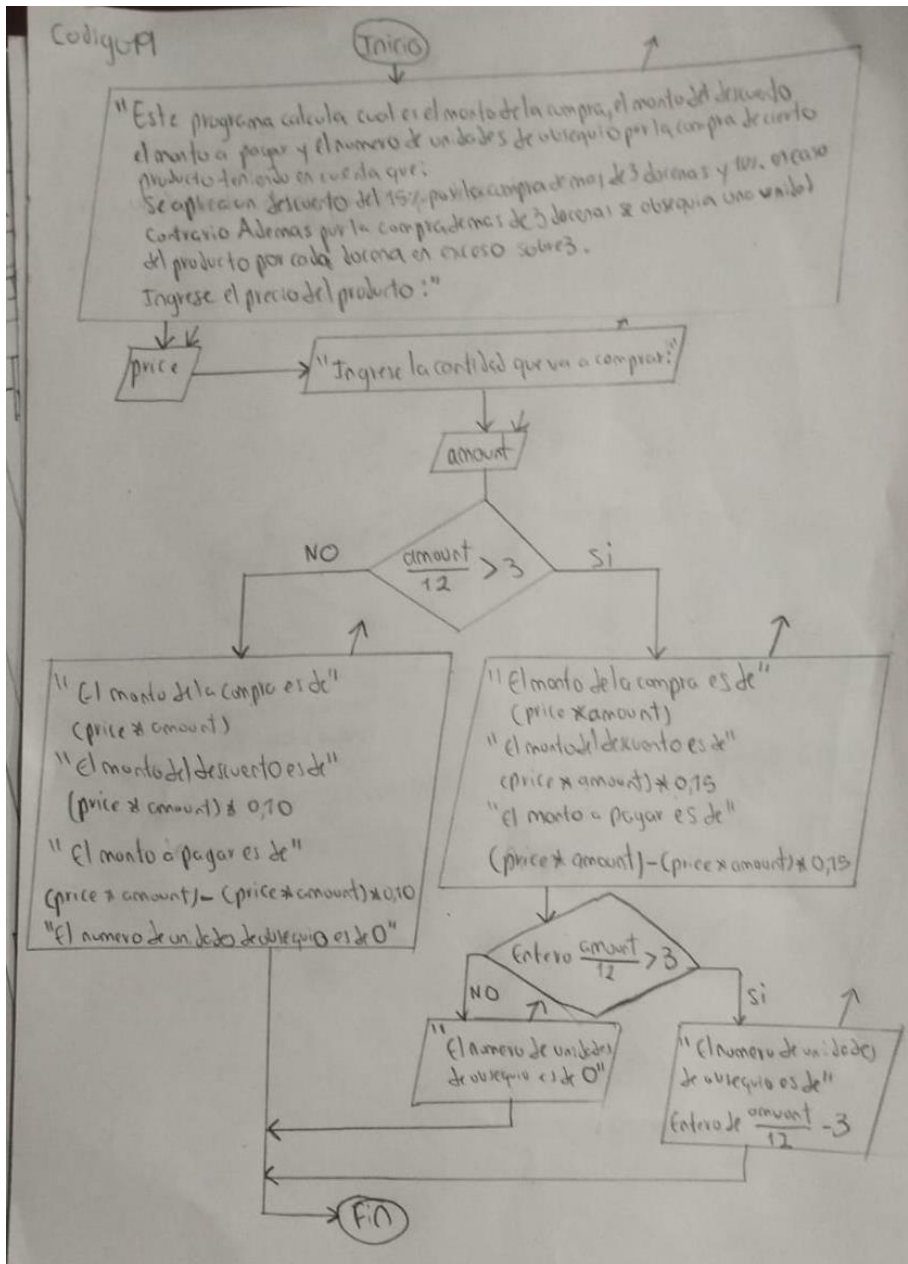
El monto de la compra es de 200000

El monto del descuento es de 30000.0

El monto a pagar es de 170000.0

El numero de unidades de obsequio es de 163

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 18:30

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa calcula cual es el monto de la compra , el monto del descuento , el monto a pagar y el numero de unidades de obsequio por la compra de cierto producto

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-Identificador : price funciona para almacenar el precio del producto

amount funciona para almacenar el numero de productos

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePrice)

(printf "Este programa calcula cual es el monto de la compra , el monto del descuento , el monto a pagar y el numero

de unidades de obsequio por la compra de cierto producto teniendo en cuenta que:

se aplica un descuento del 15% por la compra de más de 3 docenas y 10% en caso contrario

Además por la compra de más de 3 docenas se obsequia una unidad del producto por cada docena en exceso sobre 3.

Ingrese el precio del producto :")

(define price (read))

```

( printf "Ingrese la cantidad que va a comprar :")
( define amount ( read ) )
( if ( > ( / amount 12 ) 3 )
  ( begin
    ( printf "El monto de la compra es de ~a~n" ( * price amount ))
    ( printf "El monto del descuento es de ~a~n" ( * ( * price amount ) 0.15 ))
    ( printf "El monto a pagar es de ~a~n"( - ( * price amount ) ( * ( * price amount ) 0.15 ) ))
    ( if ( > ( quotient amount 12 ) 3 )
      ( printf "El numero de unidades de obsequio es de ~a" ( - ( quotient amount 12 ) 3 ) )
      ( printf "El numero de unidades de obsequio es de 0"))
    )
  (begin
    ( printf "El monto de la compra es de ~a~n" ( * price amount ))
    ( printf "El monto del descuento es de ~a~n" ( * ( * price amount ) 0.10 ))
    ( printf "El monto a pagar es de ~a~n"( - ( * price amount ) ( * ( * price amount ) 0.10 ) ))
    ( printf "El numero de unidades de obsequio es de 0")
    )))
( AnalyzePrice )

```


Código 20: Este programa determina el monto a pagar por el alquiler de un vehículo y el valor del impuesto

Enlace: <https://youtu.be/6qXhP3Akm8A>

Salida en pantalla:

```
Este programa determina el monto a pagar por el alquiler de un vehículo y el valor del impuesto.  
Ingrese los km recorridos :200  
El valor del monto a pagar por el alquiler es de 320000.0 y el valor del impuesto es de 80000.0
```

```
Este programa determina el monto a pagar por el alquiler de un vehículo y el valor del impuesto.  
Ingrese los km recorridos :10000  
El valor del monto a pagar por el alquiler es de 116720000.0 y el valor del impuesto es de 29180000.0
```

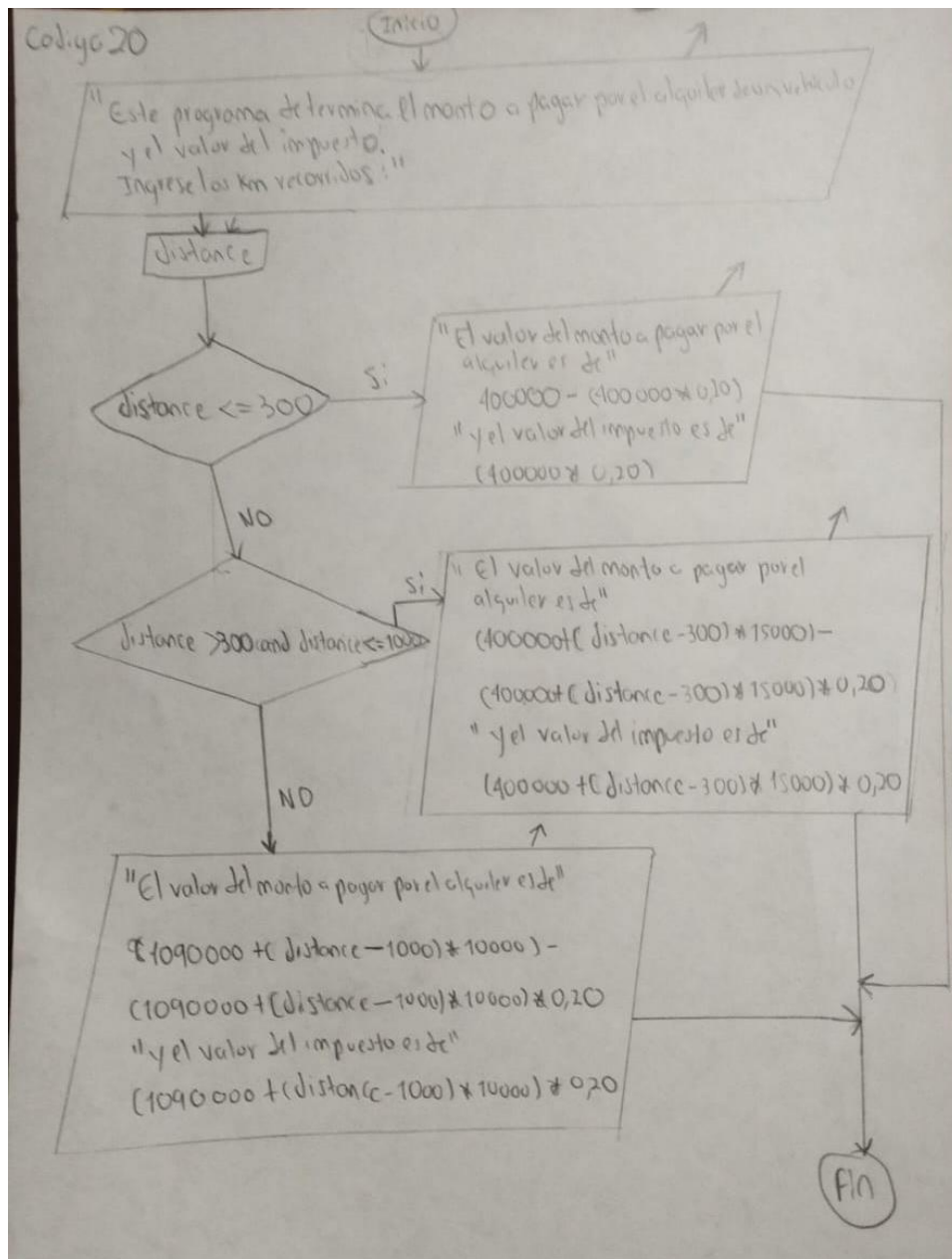
```
Este programa determina el monto a pagar por el alquiler de un vehículo y el valor del impuesto.  
Ingrese los km recorridos :1000  
El valor del monto a pagar por el alquiler es de 8720000.0 y el valor del impuesto es de 2180000.0
```

Datos entrada y salida:

Input: 200 (km recorridos)

Output: El valor del monto a pagar por el alquiler es de 320000.0 y el valor del impuesto es de 80000.0

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 19:37

-Versión de su código: 4

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este programa determina el monto a pagar por el alquiler de un vehículo y el valor del impuesto

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-Identificador : distance se utiliza para almacenar los km recorridos

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzePrice)

(printf "Este programa determina el monto a pagar por el alquiler de un vehículo y el valor del impuesto.

Ingrese los km recorridos :"

(define distance (read))

(if (<= distance 300)

(printf "El valor del monto a pagar por el alquiler es de ~a y el valor del impuesto es de ~a " (- 400000 (* 400000 0.20)) (* 400000 0.20))

(if (and (> 300) (<= 1000))

```
( printf "El valor del monto a pagar por el alquiler es de ~a y el valor del impuesto es de ~a " ( - (
+ 400000 ( * ( - distance 300 ) 15000 )) ( * ( + 400000 ( * ( - distance 300 ) 15000 ) ) 0.20 ))
```

```
( * ( + 400000 ( * ( - distance 300 ) 15000 ) ) 0.20 ))
```

```
( printf "El valor del monto a pagar por el alquiler es de ~a y el valor del impuesto es de ~a " ( - (
+ 10900000 ( * ( - distance 1000 ) 10000 )) ( * ( + 10900000 ( * ( - distance 1000 ) 10000 ) ) 0.20 ))
```

```
( * ( + 10900000 ( * ( - distance 1000 ) 10000 ) ) 0.20 ))
```

```
)))
```

```
( AnalyzePrice )
```

Código 21: Este código lee un número entero positivo, de tres cifras ($N \leq 999$ Y $N \geq 100$) y determine si el número es capicúo (es igual al revés del número original).

Enlace: <https://youtu.be/sYXtcG7UtUA>

Salida en pantalla:

```
Este código lee un número entero positivo, de
tres cifras (N <= 999 Y N>=100) y determine si el
número es capicúo (es igual al revés del número
original).
SALVEDAD: No se garantizan resultados para números no
enteros , no positivos y números que no tengan 3
cifras
Ingrese el número entero positivo de tres cifras :323
323 es un número capicuo
```

```
Este código lee un número entero positivo, de
tres cifras (N <= 999 Y N>=100) y determine si el
número es capicúo (es igual al revés del número
original).
SALVEDAD: No se garantizan resultados para números no
enteros , no positivos y números que no tengan 3
cifras
Ingrese el número entero positivo de tres cifras :456
456 es no es un número capicuo
```

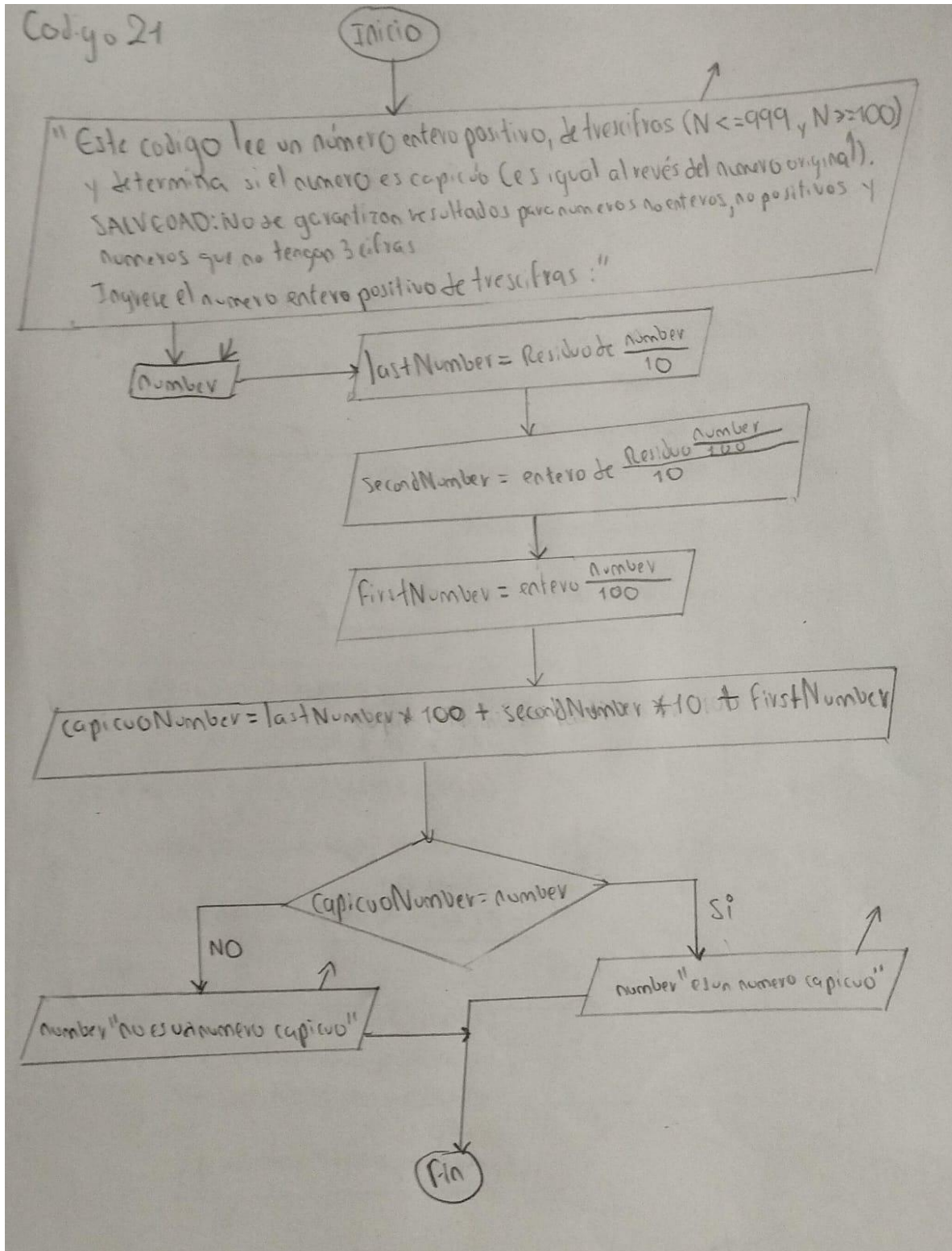
```
Este código lee un número entero positivo, de
tres cifras (N <= 999 Y N>=100) y determine si el
número es capicúo (es igual al revés del número
original).
SALVEDAD: No se garantizan resultados para números no
enteros , no positivos y números que no tengan 3
cifras
Ingrese el número entero positivo de tres cifras :101
101 es un número capicuo
```

Datos entrada y salida:

Input: 323 (número ingresado por el usuario)

Output: 323 es un número capicuo

DFD:



Código:

#lang racket

#|.

-Fecha:02/03/2024

-Hora: 19:54

-Versión de su código: 2

-Autor: Ingeniero(c) Brahian Castro Devia

-Nombre del lenguaje utilizado: Racket

-Versión del lenguaje utilizado: 8.12

-Presentado a: Doctor Ricardo Moreno Laverde

-Descripción: Este código lee un número entero positivo, de tres cifras ($N \leq 999$ Y $N \geq 100$) y determine si el número es

capicúo (es igual al revés del número original).

-Salvedades: Se garantiza el funcionamiento de este código en la versión 8.12 de Racket, no se responde por errores en otras versiones del lenguaje

-Funciona cuando se ingresan valores numéricos , de lo contrario, se expresará la violación del contrato .

-No se garantizan resultados para numeros no enteros , no positivos y numeros que no tengan 3 cifras

-Identificador : number se usa para almacenar el numero ingresado por el usuario

lastNumber se utiliza para almacenar el ultimo digito del numero de 3 cifras ingresado por el usuario

secondNumber se utiliza para almacenar el segundo digito del numero de 3 cifras ingresado por el usuario

firstNumber se utiliza para almacenar el primer digito del numero de 3 cifras ingresado por el usuario

Universidad Tecnológica de Pereira

Programa de Ingeniería de Sistemas y Computación

.|#

(define (AnalyzeNumber)

(printf "Este codigo lee un número entero positivo, de trescifras ($N \leq 999$ Y $N \geq 100$) y determine si el número es capicúo (es igual al revés del número original).

SALVEDAD: No se garantizan resultados para numeros no enteros , no positivos y numeros que no tengan 3 cifras

Ingrese el numero entero positivo de trescifras :")

```
( define number ( read ) )
```

```
( define lastNumber ( remainder number 10 ) )
```

```
( define secondNumber ( quotient ( remainder number 100 ) 10 ) )
```

```
( define firstNumber ( quotient number 100 ) )
```

```
( define capicuoNumber ( + ( * lastNumber 100 ) ( * secondNumber 10 ) firstNumber ) )
```

```
( if ( = capicuoNumber number )
```

```
  ( printf "~a es un numero capicuo" number )
```

```
  ( printf "~a es no es un numero capicuo" number )
```

```
))
```

```
( AnalyzeNumber )
```