

HITO MEJORA PROGRAMACION

MIGUEL LOZANO GARCIA
DAM

INDICE

1. Fase 1

1.1 Algoritmia.

1.2 Diagrama de flujo.

1.3 Caso de uso.

2. Fase 2

2.1 Explicación del código.

3. Fase 3

3.1 Análisis Crítico.

3.2 Paradigmas.

3.3 Validación.

FASE 1

1.1 Algoritmia.

¿Qué es un algoritmo?

Un algoritmo es una serie de operaciones que se utilizan para resolver algún problema en un tiempo limitado, este puede ser usado independientemente de su lenguaje de programación y su entorno.

¿Cómo solucionar un problema con un algoritmo?

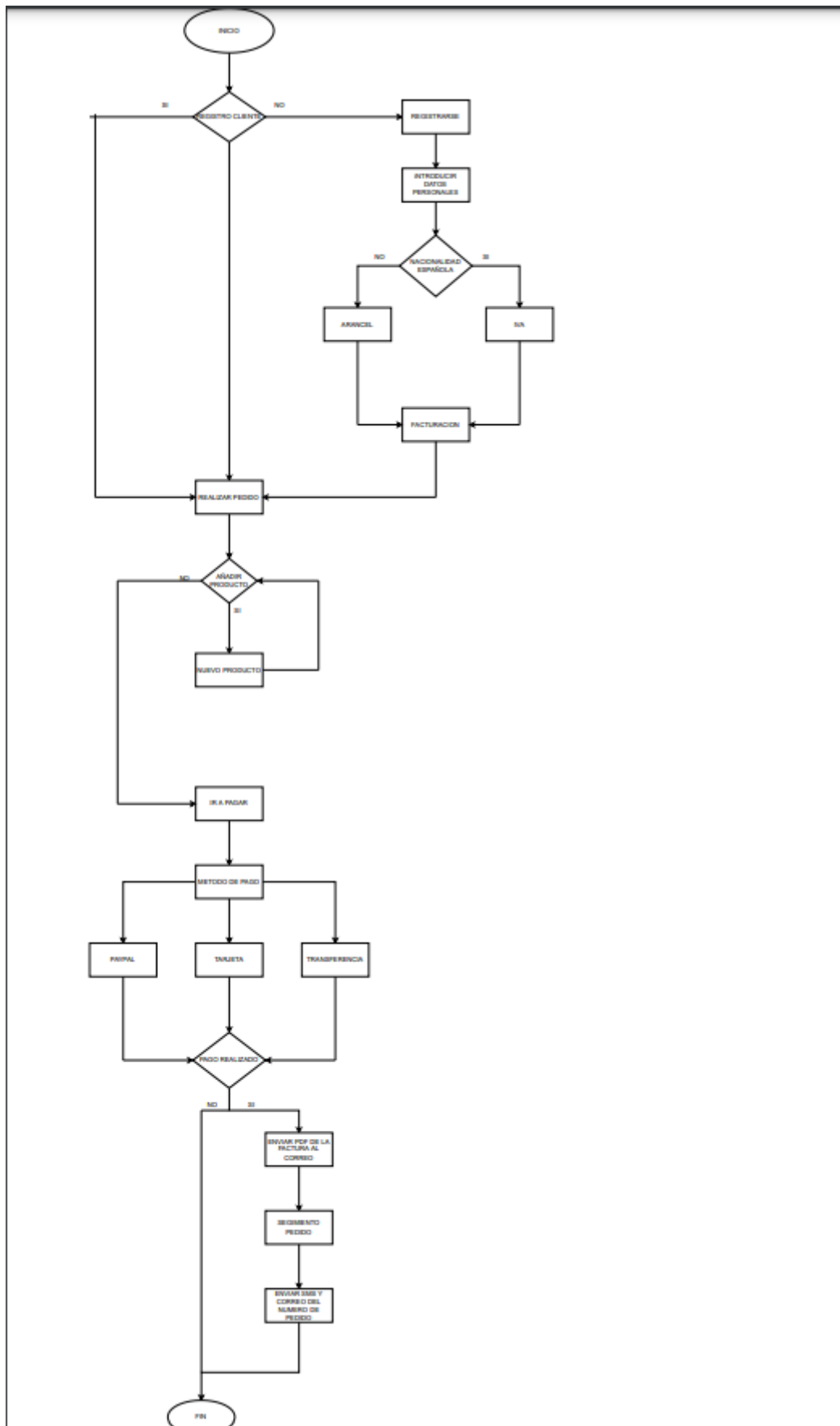
Para resolver el problema planteado debemos analizarlo detenidamente una vez hayamos realizado el análisis detallado hay que crear un procedimiento para la resolución del problema previamente analizado para a continuación picar el código en el entorno de desarrollo seleccionado hasta obtener una posible solución, la cual ejecutaremos y si en consola el procedimiento es totalmente correcto habremos llegado a la solución.

Propiedades del algoritmo.

Para que un algoritmo funcione hay que tener en cuenta lo siguiente:

El algoritmo debe tener un fin con instrucciones concretas siguiendo un orden definido, el algoritmo debe funcionar independientemente de cuales sean los datos de entrada llegando a una de las muchas soluciones posibles. Este algoritmo debe poder realizarse sin importar el lenguaje de programación y la maquina usada.

1.2 Diagrama de flujo.



En el diagrama de flujo que se ve en la imagen anterior consta de varias fases que son las siguientes: parte en el inicio, a continuación, llegamos al registro si el usuario ya está registrado previamente podrá realizar el pedido directamente, si no lo está este debe registrarse introduciendo ciertos datos y su nacionalidad para saber el impuesto que se le debe aplicar en el momento de la facturación dependiendo si su nacionalidad es de España u otra.

Una vez hecho el registro previo el usuario puede realizar su pedido entrando en un bucle añadiendo todos los productos que desee, una vez se ha terminado de añadir productos se llegará a la fase del pago.

En esta fase seleccionará un método de pago ya sea por PayPal, Tarjeta de crédito o Débito o Transferencia Bancaria e introduciremos los datos que se pidan. Luego si se ha realizado el pago correctamente se enviará la factura en formato PDF al correo del usuario con el seguimiento del pedido y un SMS junto con un correo adjuntando al cliente el número de su pedido tras este proceso el programa llegará a su fin. Si el usuario no ha realizado el pago correctamente el programa también llegará a su fin, pero sin ningún producto.

1.3 Caso de uso.

En mi aplicación el cliente tiene que darse de alta registrándose introduciendo los datos que se piden como el nombre, edad, contraseña, correo, teléfono, etc.

Una vez registrado tiene que añadir productos a través de su id y seleccionar la cantidad que desee hasta que decida ir a pagar los productos seleccionados.

En el pago el cliente seleccionará el método de pago deseado e introducirá los datos necesarios pedidos para realizar la compra.

Fase 2

2.1 Explicación del código.

```
1 #creo una clase Cliente
2 class Cliente:
3
4     def __init__(self,nombre,apellido,edad,correo,telefono,direccion,pais,contraseña): #constructor
5         self.nombre=nombre
6         self.apellido=apellido
7         self.edad=edad
8         self.correo=correo
9         self.telefono=telefono
10        self.direccion=direccion
11        self.pais=pais
12        self.contraseña=contraseña
13
14
15    def inicioPrograma(): #creamos funcion para que se registre o inicie sesion
16
17        inicio=True
18        iniciosesion=input('Escriba I para iniciar sesion o R para registrarse \n').upper()
19        while inicio == True: #bucle para que se tenga que registrar
20            reg=False
21            if iniciosesion == 'R':
22                nombre=input('Escriba su nombre: \n').upper()
23                apellido=input('Escriba su apellido: \n').upper()
24                edad=input('Escriba su edad: \n').upper()
25                correo=input('Escriba su correo electronico: \n').upper()
26                telefono=input('Escriba su numero de telefono: \n').upper()
27                direccion=input('Escriba su direccion: \n').upper()
28                pais=input('Escriba su pais: \n').upper()
29                contraseña=input('Escoja una contraseña \n').upper()
30
31                usuario=[nombre,apellido,edad,correo,telefono,direccion,pais,contraseña]
32                print(usuario)
33                reg=True
34                inicio=False
35            elif iniciosesion == 'I' and reg == False:
36                print('usuario no existe, Regístrese')
37                iniciosesion=input('Escriba R para registrarse \n').upper()
38            else:
39                print('ingrese I o R')
40                iniciosesion=input('Escriba I para iniciar sesion o R para registrarse \n').upper()
41        inicioPrograma()
42        print('\n Realize su pedido \n')
```

Al empezar el código he creado una clase llamada cliente, la cual cuenta con su constructor dentro del que se encuentran las propiedades necesarias para el

registro como el nombre, apellido, edad, correo, teléfono, dirección, país y contraseña.

Una vez terminado el constructor entero he creado una función llamada inicioPrograma la cual va a permitir que el usuario se registre o inicie sesión en la cual he creado un bucle con while en el cual la única manera de salir es registrarse ya que la aplicación es nueva y obliga a todos los usuarios a registrarse introduciendo sus datos después de crear la función la llamamos y hacemos que imprima que realice el pedido que desee.

```
43
44 #creamos una clase que herede de Cliente por si es necesario utilizar algun dato de esta clase
45 class Producto(Cliente):
46
47
48     def __init__(self,id,nombreProducto,descripcion,precio,cantidad): #constructor
49         self.id=id
50         self.nombreProducto=nombreProducto
51         self.descripcion=descripcion
52         self.precio=precio
53         self.cantidad=cantidad
54
```

He creado la clase Producto la cual hereda de Cliente ya que en más adelante se va a utilizar el correo y el teléfono del usuario, a continuación, he creado el constructor de la nueva clase con las propiedades id, nombreProducto, descripción, precio y cantidad.

```
55 #muestra los productos que se pueden comprar
56 print('A','mesa','mesa redonda',20,1)
57 print('B','silla','silla madera',8,4)
58 print('C','pantalon','pantalon corto',12,1)
59 print('D','calcetines','calcetines negros',7,6)
60
```

He añadido algunos productos al azar que pueden ser seleccionados mediante el id de este (A, B, C, D)

```
61 #creamos una funcion que usaremos luego para que nos imprima el producto deseado
62 def seleccionProducto(self):
63     print(f'Ha añadido a la cesta el producto {self.nombreProducto} cuyo identificador es "{self.id}" se trata de {self.descripcion}, tiene un coste de {self.precio} euros por unidad ')
64     cantidad=int(input('¿cuantas unidades desea? \n'))
65     if cantidad <= self.cantidad:
66         print('unidades añadidas')
67     else:
68         print('no disponemos de suficientes existencias')
69
70
```

Creo una nueva función llamada seleccionProducto dentro de la clase Producto que más adelante llamare para que imprima el producto que el cliente desea y luego indique la cantidad que desea usando un condicional para que imprima el

número de unidades deseadas o si se excede de las existencias que le imprima que no disponemos de suficientes existencias.

```
71 A=Producto('A','mesa','mesa redonda',20,1)
72 B=Producto('B','silla','silla madera',8,4)
73 C=Producto('C','pantalon','pantalon corto',12,1)
74 D=Producto('D','calcetines','calcetines negros',7,6)
75
```

En esta imagen he creado una variable con el mismo nombre que el id del producto para poder usarlo en la siguiente imagen

```
76 #utilizo un bucle para que pueda seguir comprando mas productos
77 continuar=True
78 while continuar:
79     selecProducto=input('Seleccione un producto de los recien mostrados mediante su identificador o pulse P para ir a pagar ')
80     if selecProducto == 'a' or selecProducto == 'A':
81         print(A.seleccionProducto())
82         continuar
83     elif selecProducto == 'b' or selecProducto == 'B':
84         print(B.seleccionProducto())
85         continuar
86     elif selecProducto == 'c' or selecProducto == 'C':
87         print(C.seleccionProducto())
88         continuar
89     elif selecProducto == 'd' or selecProducto == 'D':
90         print(D.seleccionProducto())
91         continuar
92     elif selecProducto == 'p' or selecProducto == 'P':
93         print('solo admitimos pagos con tarjeta de debito o credito')
94         int(input('introduzca su numero de tarjeta \n'))
95         print('se le cargara el importe de su compra a la tarjeta recien introducida')
96         gmail=input('introduzca su correo electronico: ')
97         print(f'su gmail es {gmail}')
98         try:
99             telf=(int(input('Introduzca su número de telefono: ')))
100         except:
101             print('Has introducido un carácter que no es un número, introduzca un número para poder continuar')
102             telf=(int(input('Introduzca su número de telefono: ')))
103         print(f'se enviara la factura como pdf a {gmail}')
104         print(f'no olvide revisar los mensajes del numero {telf} y el correo {gmail} donde le llegara un seguimiento de su pedido')
105         print(f'Gracias por su compra')
106         continuar=input('¿Desea hacer una nueva compra?(si/no) ') == 'si'
107
108     else:
109         print('digito no valido')
110         continuar=input('¿Desea añadir mas prouctos?(si/no) ') == 'si'
111
```

Para que el cliente pueda comprar todos los productos que desee he creado un bucle con while en el que cada vez que se escriba el id del producto deseado llamo a la función creada anteriormente llamada selecciónProducto junto con la variable del id creada del cual se saldrá pulsando la tecla 'p' para ir a pagar donde te pedirá el número de tarjeta, el correo electrónico y su número de teléfono con un try para que si escribe un digito que no sea un numero se lo comunique y le permita volver a intentarlo. A continuación, le imprimirá que se le enviará un PDF al Gmail anteriormente escrito y que revise su número y su correo para el seguimiento de su pedido.

Para terminar, le preguntara al cliente si desea hacer una compra nueva en la cual si le damos a si volveremos al bucle y si ponemos no llegaremos al fin de la aplicación.

FASE 3

3.1 Análisis crítico.

La aplicación creada consiste en una aplicación para comprar productos en la que un usuario se registrará introduciendo los datos que se le pida, seleccionará tantos productos como desee y la cantidad de estos, una vez hecho esto se le dirigirá al cliente el apartado del pago donde introducirá los datos de su método de pago y realizara su pedido.

3.2 Paradigmas de programación.

- He utilizado como paradigma la Programación Orientada a Objetos (POO) ya que he utilizado para mi aplicación dos clases ambas con sus respectivos constructores una clase padre (Cliente) y otra que hereda de esta (Producto), por lo tanto, también he utilizado herencia.
- También encontramos paradigma imperativo ya que he utilizado bucles con while y condicionales con if. Se trata de un código con bastantes líneas y ciertos comentarios en él.

3.3 Validación.

En el caso que al inicio de sesión el usuario pulse i la aplicación le va a decir que no existe el usuario que tiene que registrarse.

En el momento de seleccionar productos se seleccionan por si id (A, B, C, D) o P para pagar si escribes cualquier otro carácter te indicara que se trata de un dígito no válido y que si desea añadir más productos si escribes si te dirá que escribas el id del producto que desees si escribes no finalizara el programa.

Si al elegir el producto selecciona más unidades de las existentes el propio programa se lo indicara por consola.

En el momento de pagar si al escribir el número de la tarjeta no seleccionas un número el programa dará error ya que no es un int y en el teléfono al usar el try si no escribes un número tendrás un intento nuevo imprimiendo que el carácter introducido no es un número y si lo introduces erróneo de nuevo dará un fallo en el programa.