

# **Universidad Nacional Autónoma de México**

## **Facultad de ingeniería.**

### **Estructura de datos y algoritmos 2**

**Prof Gerardo Tovar Tapia**

**Equipo 1**

#### **Entrega del proyecto 2.**

##### **DESCRIPCIÓN DEL PROYECTO:**

En este proyecto se harán registros de materias, donde para poder acceder a este registro es necesario el numero de cuenta y una clave única por alumno. Se creará un menú donde tenga dichas opciones (Login y salir), dentro de login, podrás ingresar tu numero de cuenta y clave, si alguna de estas es incorrecta entonces se mandará un mensaje de que la clave o el numero de cuenta es incorrecto. Una vez ingresando dichos datos se mostrará la información del alumno (Nombre, Numero de Cuenta y promedio), con su respectivo menú que contendrá las siguientes opciones:

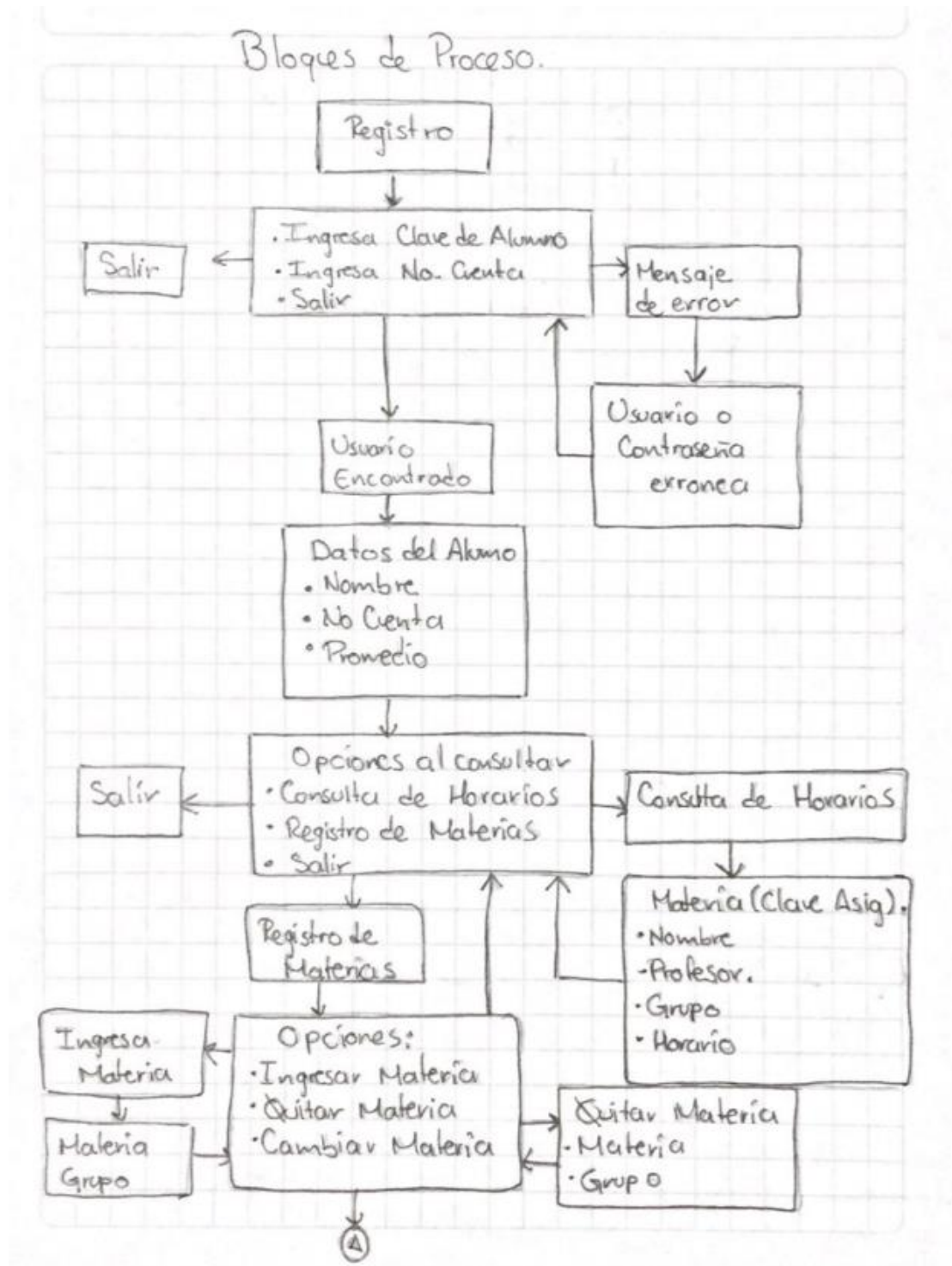
- Consulta de Horarios: En esta sección se mostrará al alumno los horarios disponibles, con su respectiva información; Nombre de la Materia, Clave, Profesor, Grupo, Horario y Vacantes.
- Registro de materias: Dentro de esta sección se mostrará otro menú donde pondremos las siguientes opciones:
  1. Ingresar Materia: El usuario ingresará nombre de la Materia y Grupo, para darla de alta en su horario actual
  2. Quitar Materia: El usuario ingresará nombre de la Materia y Grupo, para así removerla de dicho horario.
  3. Cambiar Materia: El usuario ingresará Nombre y Grupo que desea remover y posteriormente Nombre y grupo nuevos, para así actualizar su información.
- Salir: Permitirá al usuario salir de la aplicación guardando toda la información antes de cerrar.

## Diagrama de flujo:

(Debido a la extensión del diagrama se anexa en un documento pdf aparte.)

→ [Diagrama.pdf](#)

## Diagrama de bloques:





Avance del programa documentado.

→Pro.py

```

#Definición de la clase alumno.
"""La clase alumno es la que contendrá toda la información de los alumnos
desde su nombre, número de cuenta, clave, materias y promedio."""
class Alumno:
    nombre=''
    numCuenta=''
    claveAcceso=''
    materias=[]
    calificaciones=[]
    promedio=0
    #Constructor de la clase Alumno.
    """Recibe nombre,número de cuenta y las materias con sus calificaciones"""

    def __init__(self,n,nu,ma,cal):
        self.nombre=n
        self.numCuenta=nu
        self.materias=ma
        self.calificaciones=cal

    #Agrega una materia al arreglo de materias del alumno
    def addMaterias(self,mat):
        self.materias.append(mat)

    #Recibe el arreglo de calificaciones y después de calcularlo retorna el promedio.
    def getPromedio(self,cal):
        if len(mat)!=0:
            for i in range(len(mat)):
                p+=(int(cal[i]))/(len(mat))
            return p

    #Muestra la información del alumno-->nombre, número de cuenta y su promedio.
    def GetAlumno(self):
        print("Nombre: ",self.nombre)
        print("Numero de cuenta: ",self.numCuenta)
        print("Promedio: ",self.promedio)

def Capturar():
    total=[]
    #Abre el archivo(al estar terminado será el registro de alumnos)
    doc=open("prueba.txt")
    #Iguala linea a la lectura de una linea del documento.
    linea=doc.readline()
    #Mientras linea contenga algo dentro se ejecutara todo el código.
    while linea!='':
        if linea!=None:
            #Convierte linea en un arreglo de cadenas llamado "palabras"
            palabras = linea.split()
            #Si el tamaño de la lista "palabras" es mayor a 0 accede al if.
            if len(palabras)>0:
                """Aquí busca que el primer elemento de la lista sea "Nombre:" si es así
                comienza a conseguir valores para crear un alumno el cual se guardará en
                otra lista llamada "total" la cual contendrá todos los alumnos que están
                dentro del documento que contiene a los alumnos"""

```

```

        if palabras[0]=='Nombre:':
            nom=""
            mat=[]
            cal=[]
            for i in range(1,len(palabras)):
                #Se guarda el nombre del alumno sacado del documento en una cadena 'nom'
                nom+=palabras[i]
                nom+=" "

            linea=doc.readline()
            palabras=linea.split()
            #Guarda el numero decuenta sacado del documento en una variable 'cuenta'
            cuenta=palabras[1]
            linea=doc.readline()
            palabras=linea.split()
            #Guarda todas las materias sacadas del documento y las gaurda en una lista 'mat'
            for i in range(1,len(palabras)):
                mat.append(palabras[i])

            linea=doc.readline()
            palabras=linea.split()
            #Guarda todas las calificaciones sacadas del documento y las guarda en una lista 'cal'
            for i in range(1,len(palabras)):
                cal.append(palabras[i])

            #Crea un alumno con los parametros obtenidos anteriormente y los guarda en una lista 'total'
            total.append(Alumno(nom,cuenta,mat,cal))

        #Continua leyendo liena a linea
        linea=doc.readline()
    #Retorna la lista con todos los alumnos localizados en el documento.
    return total

#Main
#Se llama la funcion capturar para que 'a' contenga a todos los alumnos registrados.
a=Capturar()
#Se inicializa 'acceso' en falso
acceso=False
"""Pide los datos para poder acceder, mientras estos datos sean diferentes a cualquiera
de los alumnos dentro de la lista 'a' no se podra acceder."""
while acceso==False:
    print("Sistema de inscripcion (primer semestre)")
    count=input("\nIntrordusca tu número de cuenta: ")
    key=input("Introduce tu clave de acceso: ")
    for i in a:
        #Dentro de toda la lista de alumnos busca la coincidencia del número de cuenta.
        if i.numCuenta==count:
            #Una vez encontrada la cuenta se comprueba si la cuenta es correcta.
            if i.claveAcceso==key:
                acceso=True
                #Llama la función GetAlumno() para mostrar los datos del usuario accedido.
                i.GetAlumno()
            else:
                print("La cuenta con clave no coinciden")

#Mientras el acceso sea verdadero podras navegar por todo el menú y las opciones del registro.
while acceso==True:
    #Impresión del menú
    print("**Consultar Horarios...(1)")
    print("**Registrar Materias...(2)")
    print("**Salir...(0)")
    op=int(input("¿Qué deseas hacer?"))
    #Eleccion de opcion
    if op==1:
        print("holi")
    elif op==2:
        print("no holi")
    else:
        #Opcion para salir. Iguala 'acceso' a falso para poder salir del ciclo while
        #y el programa finalice
        print("Saliendo...")
        acceso=False
#Fin del programa

```

Nota:

El programa usa un archivo 'prueba.txt' el cual esta anexo.

### Cronograma de entregas:

Entrega 2 11-Nov-2017	<b>Documento presente con:</b> <ul style="list-style-type: none"><li>• Mayor descripción del proyecto y sus procesos.</li><li>• Diagrama de flujo esperado.</li><li>• Diagrama de bloques esperado.</li><li>• Código documentado con el primer avance</li><li>• Primer Cronograma.</li></ul>
Entrega 3 19-nov-2017	<b>Documento con:</b> <ul style="list-style-type: none"><li>• En caso de reajustes:<ul style="list-style-type: none"><li>◦ Describir ajustes</li><li>◦ Diagramas con reajustes(flujo y procesos)</li></ul></li><li>• Interfaz gráfica, mayormente terminada.</li><li>• Código documentado mínimo con los procesos:<ul style="list-style-type: none"><li>◦ Consultar horarios</li><li>◦ Menú de Registrar materias</li></ul></li><li>• Nuevo cronograma(Actualizado).</li></ul>
Entrega 4 26-Nov-2017	<b>Documento con:</b> <ul style="list-style-type: none"><li>• Interfaz gráfica completa.</li><li>• Código documentado todos los procesos implementados.</li><li>• Documentos a usar completos.</li><li>• Nuevo cronograma(Actualizado).</li></ul>
Entrega 5 3-Dic-2017	<b>Corrección de errores del proyecto.</b> <b>Nuevo cronograma de ser necesario.</b>
Entrega Final 8-Dic-2017	<b>Entrega de Proyecto</b>

Nota: Este cronograma está ajustado a las fechas de entrega asignadas por el profesor, en cualquier momento el equipo podrá modificarlo y adelantar la fecha de entrega dependiendo del progreso y avance del proyecto .