



Calificación

Examen: ____ / 12

Deberes: ____ / 4

Lección: ____ / 4

Total: ____ / 20

EVALUACIÓN DEL PRIMER PARCIAL - Examen

MATERIA: ALGEBRA LINEAL

NOMBRE DEL ESTUDIANTE:.....

CARRERA.....MODALIDAD :

FECHA:.....

NOMBRE DOCENTE: Ing. Kleber Loayza A. MSc.

INSTRUCCIONES

Lea detenidamente antes de responder las siguientes preguntas o afirmaciones, utilice letra clara y evite los manchones. Recuerde que no está permitido el uso de celulares en un examen ni lápiz. Al lado de cada pregunta consta el puntaje de calificación de cada una

RECOMENDACIONES.

- TENER TODO LISTO
- ESTAR TRANQUILO, CONFIAR EN SI MISMO Y EN LO QUE SABE
- LEER ATENTAMENTE A CADA PREGUNTA
- ENTENDER LO QUE PIDE LA PREGUNTA
- LEER LAS ALTERNATIVAS
- RESPONDER PRIMERO TODAS LAS PREGUNTAS QUE SABE
- VOLVER POR LAS QUE DEJO PENDIENTE
- NO DEJAR DE RESPONDER NINGUNA PREGUNTA
- CONTROLAR EL TIEMPO
- EVITE LAS DUDAS FINALES
- NO INTENTE PREGUNTAR A OTRO

OBJETIVO

Evaluar los conocimientos logrados en la materia de Redes de Acceso 1 obtenidos durante el transcurso de las clases presenciales dictadas y las actividades realizadas para mejor comprensión de la materia.

MISIÓN

Somos una universidad de grado y posgrado comprometida con la calidad de la educación superior a través de la formación de profesionales competentes para un entorno global, teniendo como plataforma principal, en el proceso educativo, el uso de las tecnologías de la información y las comunicaciones, creando y compartiendo nuevo conocimiento mediante la investigación científica que genera desarrollo para Guayaquil y el país.

VISIÓN

Para el 2019 seremos una universidad internacionalizada, de vanguardia en el desarrollo del emprendimiento y la innovación, trascendiendo en la dinámica del progreso de Guayaquil y el país.

VALORES INSTITUCIONALES

- Lealtad
- Compromiso
- Disciplina
- Solidaridad
- Integridad
- Responsabilidad Social
- Respeto Al Medio Ambiente.

I. Tema 1: Diagrama de Clase

Realizar un diagrama de clase del siguiente problema:

La reservación de una habitación en un hotel, donde un cliente se contacta con el recepcionista del hotel, mediante una llamada telefónica, generando un detalle de la reservación. Considerar que parte de la reservación hay que tomar la opción si cuenta con garaje.

Se espera observar:

Asociaciones, Herencia, Multi-Herencia, Cardinalidad.

Tip:

Posibles Objetos:

Hotel, Reservación, Detalle Reservación, Cliente, Recepcionista, Coche, Garaje.

II. Tema 2: Objetos Semejantes

Dada la siguiente porción de código, ¿En qué líneas hay errores?

```
3 class Point
4     def __init__(self, x, y):
5         self.move(x,y)
6     def move(x, y):
7         self.x=x
8         self.y=y
9 #-----Main-----
10 p1= Point(0,0)
11 p2= Point(0,0)
12 print("Punto 1")
13 print(p1.x,p1.y)
14 p1.move(p1.x)
```

- a) 4, 11 y 13
- b) 13, 7 y 4
- c) 3, 6 y 8
- d) 6, 3 y 14
- e) 10, 14 y 4

III. Tema 3: Herencia Básica

Dada la siguiente porción de código, donde la clase Supplier hereda de Contact, ¿En qué líneas hay errores?

```
1 class Contact:
2     def __init__(self,name, email):
3         self.name=name
4         self.email=email
5         self.agregarcontacto.append(self)
6
7 class Supplier():
8     def order():
9         return "Imprimir la orden de {}".format(self.name)
10
11
12 contacto=Contact("Juan","jlopez@gmail.com")
13 proveedor=Supplier("Gustavo","glopez@gmail.com")
14
15 #-----Impresion del objeto Contacto-----
16 print("Contacto:")
17 print("Nombre: ",contacto.name)
18 print("Email: ",contacto.email)
19 print("Orden: ",contacto.order())
20 #-----Impresion del objeto Proveedor-----
21 print("Proveedor:")
22 print("Nombre: ",proveedor.name)
23 print("Email: ",proveedor.email)
24 print("Orden: ",proveedor.order())
```


V. Tema 5: Multi-Herencia

Dado el siguiente código:

```
1 class Baseclass:
2     def call_me(self):
3         print("Gabriel",end="-")
4 class Leftclass(Baseclass):
5     def call_me(self):
6         super().call_me()
7         print("Pedro",end="-")
8 class Rightclass(Baseclass):
9     def call_me(self):
10        super().call_me()
11        print("Luis",end="-")
12 class Subclass(Leftclass,Rightclass):
13     def call_me(self):
14         super().call_me()
15         print("Mario",end="-")
16 ##### Main #####
17 s = Subclass()
18 s.call_me()
```

Indique que imprime en su ejecución:

- a) Gabriel-Pedro-Luis-Mario-
- b) Gabriel-Luis-Pedro-Mario-
- c) Mario-Luis-Gabriel-Pedro-
- d) Mario-Luis-Pedro-Gabriel-
- e) Luis-Pedro-Gabriel-Mario-
- f) Ninguno de las anteriores.

VI. Tema 6: Polimorfismo

Dado el siguiente código:

```
1 class AddressHolder:
2     def __init__(self, street='', city='', state='', code='', **kwargs):
3         super().__init__(**kwargs)
4         print("Welcome to Address Holder class")
5         self.street=street
6         self.city=city
7         self.state=state
8         self.code=code
9
10 class Contact():
11     def __init__(self, name='', email='',**kwargs):
12         super().__init__(**kwargs)
13         print("Welcome to Contact class")
14         self.name=name
15         self.email=email
16
17 class Friend(AddressHolder,Contact):
18     def __init__(self, phone, **kwargs):
19         super().__init__(**kwargs)
20         print("Welcome to Friend class")
21         self.phone=phone
22
23 frien_1 = Friend('0994558746')
```

¿Cuál es la salida?

- a) Address Holder- Contact-Friend-
- b) Contact-Address Holder-Friend-
- c) Friend-Address Holder- Contact-
- d) Muestra mensaje de error
- e) Ninguna de las anteriores

VII. Tema 7: Clases Abstractas

Dado el siguiente código:

```
1 import abc
2
3 class Medialoader(metaclass=abc.ABCMeta):
4     abstractmethod
5     def play(self):
6         pass
7     @abc.abstractproperty
8     def ext(self):
9         pass
10
11 class Mp3(Medialoader):
12     pass
13     def play():
14         print("Dispositivos mp3")
15     ext='mp3'
16
17 class Wav(Medialoader):
18     pass
19     def play(self):
20         print("Dispositivos wav")
21
22 w1 = Wav()
23 m1 = Mp3()
24 m1.play()
```

¿Cuáles son las líneas que producen error?

- a) 3 y 19
- b) 4 y 19
- c) 3 y 21
- d) 4 y 21

VIII. Tema 8: Manejo de Excepciones

8.1) ¿Cuál de las siguientes porciones de código, produce un error?

1)

```
1 list = [12,56,31]
2 list.append(56)
3 print(list)
```

2)

```
5 d={'a':'azul', 'b':'amarillo'}
6 print(d['c'])
```

3)

```
5 a = 2 + 8
6 b = 6+1
7 print(a+b+c)
```

- a) 1 y 2
- b) 1 y 3
- c) 2 y 3
- d) Ninguna de las anteriores

8.2) Al generar la siguiente porción de código:

```
10 def sumnumbers(av, bv):
11     return av + bv
12
13 nombre_a= 'Juan'
14 edad_b = 5
15 print(sumnumbers(nombre_a,edad_b))
```

Me arroja la siguiente salida:

```
Traceback (most recent call last):
  File "q7.py", line 16, in <module>
    print(sumnumbers(nombre_a,edad_b))
  File "q7.py", line 11, in sumnumbers
    return av + bv
TypeError: can only concatenate str (not "int") to str
```

En el siguiente cuadro escriba el código otra vez, incorporando el manejo de excepciones.

FIRMA DEL ESTUDIANTE

FIRMA DEL DOCENTE