



UNIVERSIDAD ABIERTA INTERAMERICANA
FACULTAD DE TECNOLOGÍA INFORMÁTICA

MATERIA: PROGRAMACIÓN II

1er Parcial

Teórico y Práctico

Alumno:

Fecha:

Tema 1

Comisión – Localización - Turno:

Práctica:

Teoría:

Nota:

Temas para evaluar: Paradigmas: Estructurado y POO, introducción al lenguaje python.

Objetivos:

Comprender las pautas del proceso para la creación de software desde la perspectiva de la Ing. SW.
Comprender cómo se aplican las técnicas del paradigma estructurado y POO en el lenguaje python

Modalidad: Parcial domiciliario

Requisitos para aprobar: Para que el parcial esté aprobado el alumno deberá tener correctamente desarrolladas el 60% de la teoría y resuelto el ejercicio práctico.

Tiempo:

Recomendaciones:

- a) Lea todo el parcial antes de comenzar a responder.
- b) Desarrolle una redacción clara y precisa contestando lo que la pregunta requiere.
- c) Observe la ortografía ya que la misma es parte del parcial.
- d) Si considera que no comprende alguna consigna antes de comenzar consulte a su profesor.

Notas: Las preguntas en las que se seleccionen opciones se deberá optar solo por una de las posibilidades. La indicación se efectuará con una X sobre su lateral izquierdo, será considerada nula si presenta tachaduras o enmiendas.

Las preguntas que solicitan justificación serán consideradas válidas si poseen la misma correctamente.

Las preguntas de múltiples posibilidades y verdadero / falso restan 0.50 puntos en caso de estar mal contestadas. En las preguntas verdadero / falso se debe tachar la opción incorrecta.

(*) la cifra entre paréntesis en cada pregunta es la cantidad de puntos sobre 100.

1. Indique tres diferencias sustanciales al momento de diseñar un programa utilizando el paradigma estructurado y paradigma orientado a objetos. (10)
2. ¿Por qué decimos que python es un lenguaje semi-interpretado y con tipado dinámico? (10)
3. ¿Qué estructuras condicionales permite el lenguaje python? Ejemplifique con una porción de código. (10)
4. ¿Qué estructuras repetitivas permite el lenguaje python? Ejemplifique con una porción de código. (10)



UNIVERSIDAD ABIERTA INTERAMERICANA
FACULTAD DE TECNOLOGÍA INFORMÁTICA

MATERIA: PROGRAMACIÓN II

1er Parcial

Teórico y Práctico

5. ¿Cuáles son las diferencias entre listas, tuplas y diccionarios en python? (10)

6. ¿Cómo se declara una clase? Ejemplifique (10)

7. ¿Cómo declaro una propiedad? Ejemplifique (10)

8. ¿Qué es el Duck typing? (10)

9. Ejemplifique cómo se puede implementar polimorfismo con python (10)

10. ¿Cuál es la diferencia entre los métodos init y new? (10)



UNIVERSIDAD ABIERTA INTERAMERICANA
FACULTAD DE TECNOLOGÍA INFORMÁTICA

MATERIA: PROGRAMACIÓN II

1er Parcial

Teórico y Práctico

PRÁCTICA

1. Escribir una clase **Personaje** que contenga los atributos **vida**, **posicion** y **velocidad**, y los métodos **recibir_ataque**, que reduzca la vida según una cantidad recibida y escriba un mensaje por consola si la vida pasa a ser menor o igual que cero, y **mover** que reciba una dirección y se mueva en esa dirección la cantidad indicada por velocidad. (20)
2. Escribir una clase **Soldado** que herede de **Personaje**, y agregue el atributo **ataque** y el método **atacar**, que reciba otro personaje, al que le debe hacer el daño indicado por el atributo **ataque**. (20)
3. Escribir una clase **Campesino** que herede de **Personaje**, y agregue el atributo **cosecha** y el método **cosechar**, que devuelva la cantidad cosechada. (20)
4. Los soldados pueden atacar a soldados o campesinos. No se debe permitir que un campesino ataque a otro campesino o a un soldado. (10)
5. Genere código python para probar la solución propuesta con la cantidad de personajes que crea conveniente y la prueba del mensaje cuando los personajes tengan vida 0 o se les indique moverse, atacar o cosechar. (30)

Firma del Alumno