



UNIVERSIDAD ESTATAL A DISTANCIA
Vicerrectoría de Docencia
Escuela de Ciencias Exactas y Naturales
Cátedra Registros de Salud
03564 Lógica y Algoritmos para Sistemas
de Información en Salud



**INSTRUCCIONES PROYECTO 1: SISTEMA DE SEGUIMIENTO DE SIGNOS VITALES
EN PACIENTE**

PERIODO ACADÉMICO: I CUATRIMESTRE 2025

VALOR: 40%

Objetivo: Evaluar críticamente el uso y la selección de tipos de datos en la implementación de estructuras y variables para optimizar la manipulación de datos demostrando las habilidades de pensamiento crítico y actitud de indagación para el planteamiento y la resolución de problemas por medio de la herramienta Python y evaluación del conocimiento en Microsoft Teams.

Consideraciones generales:

1. El proyecto se entrega únicamente por medio de la plataforma de aprendizaje AprendeU a partir del **10 de marzo de 2025 desde las 00:00 horas al 16 de marzo de 2025 a las 23:55 horas**, no espere a realizar la entrega hasta el último día, ya que no se recibirán trabajos extemporáneos, causas justificadas deben apegarse al Artículo 4, inciso i) del Reglamento General Estudiantil. Recuerde que el proyecto se hace de forma individual.
2. Antes de realizar la entrega, verifique que sea el documento correcto, de lo contrario perderá los puntos de la asignación. Se calificará el documento que la persona estudiante adjunte, si adjunta algún archivo que se encuentre dañado o que no corresponde al instrumento a evaluar, no se le calificará y perderá los puntos, tampoco se calificarán los archivos que queden en la bandeja de borrador, verifique y acepte las condiciones de la plataforma para hacer el envío. Recuerde revisar todos los requerimientos anteriores antes de entregar el proyecto.
3. En caso de que la conexión a internet o el servicio eléctrico falle deberá presentar una certificación del proveedor que indique que efectivamente tuvo una interrupción del servicio en el día y hora de la entrega, la misma deberá ser presentada en un plazo no mayor a los

- 3 días posteriores a la entrega de esta tarea de lo contrario no se le aceptará y perderá los puntos de la asignación.
4. Antes de iniciar con el proyecto recuerde que debe haber leído el módulo de aprendizaje de la Unidad 1 y 2 y realizar los ejercicios de autoevaluación.
 5. La persona estudiante que tiene poca accesibilidad de internet debe coordinar con el centro universitario al que pertenece el uso de una máquina para que coordinen con anticipación, ejecuten el proyecto y la entreguen en la fecha establecida.
 6. Si tiene dudas sobre algún asunto relacionado con el proyecto, utilice el foro de consultas que se encuentra disponible todo el cuatrimestre en la Unidad 0 y procure realizar sus preguntas con el tiempo suficiente para brindarle una atención oportuna.
 7. Se les recuerda a todos los estudiantes que la UNED pone a disposición a través del entorno estudiantil el Office 365 de forma gratuita.
 8. La persona estudiante debe tener instalada la aplicación de Microsoft Teams versión escritorio para unirse a la sesión de evaluación del conocimiento y compartir pantalla al tutor.

Por lo tanto, para el desarrollo del proyecto deberá estructurarla de la siguiente manera:

Para el desarrollo del proyecto deberá apegarse al lineamiento de presentación de trabajos escritos, disponible en la Unidad 0, apegándose al lineamiento de presentación de trabajos escritos y estructurarla de la siguiente manera:

1. Formato

Para cumplir con el logro de aprendizaje la persona participante debe apegarse al lineamiento de presentación de trabajos escritos y cumplir con cada una de las indicaciones.

Formato del documento	
Dimensiones del documento	Normales
Fuentes y tamaños	Arial, tamaño 11
Numeración de las páginas	Insertada en la parte inferior de la página, centrado en números arábigos. La portada no lleva numeración.
Márgenes	Normales
Títulos y subtítulos	Títulos en negrita, tamaño 14 Subtítulos en negrita, tamaño 12
Formato del archivo	.docx es decir en Word, .py es decir archivo Python

Formato del documento	
Estructura en el nombre del archivo	Apellido1Apellido2Nombre_Proyecto1, todo pegado sin espacios para ambos archivos.
Alineación	Justificado
Color	Automático (negro)

2. Portada

Para cumplir con el logro de aprendizaje la persona participante debe apegarse al lineamiento de presentación de trabajos escritos y cumplir con cada una de las indicaciones. Debe distribuir toda la información a lo largo de la página tal y como se muestra en el Lineamiento para la presentación de trabajos escritos (ver ejemplo, *lo que se encuentra en mayúscula se debe de colocar en mayúscula y viceversa*).

- Encabezado
 - NOMBRE DE LA UNIVERSIDAD
 - NOMBRE DE LA VICERRECTORÍA A LA QUE PERTENECE LA ESCUELA
 - NOMBRE DE LA ESCUELA
 - NOMBRE DE LA CARRERA
 - NOMBRE DE LA CÁTEDRA
 - Logo de la UNED en formato png.
- TÍTULO DE LA TAREA
- Asignatura y código
- Nombre del estudiante
- Cédula de identidad
- Nombre del Tutor
- Fecha de entrega
- CUATRIMESTRE Y AÑO

3. Tabla de contenido

Apegándose al lineamiento de presentación de trabajos escritos.

- El mismo debe estar disponible en el documento de forma automática.
- La misma debe de contar con la utilización de al menos tres estilos de forma correcta.
- Recuerde que debe actualizar el formato de tamaño y letra cada vez que incluya algún apartado de su trabajo.

Apartado	Estilo por utilizar
Desarrollo	Título 1
Ejecución de programas de Python: ➔ Función 1 ➔ Función 2 ➔ Función ... ➔ Función n	Título 2
Rúbrica de calificación	Título 3

4. Desarrollo

Para cumplir con el logro de aprendizaje, la persona participante debe apegarse al lineamiento de presentación de trabajos escritos y cumplir con cada una de las indicaciones:

Nota: Debe documentar para cada uno de los programas los resultados de ejecución obtenidos de las funciones en Python por medio de una descripción y sus capturas de pantalla e incluirlo en el documento de Word e incluir el archivo adjunto “.docx” y “.py” del programa completo que usted ha desarrollado en la plataforma de aprendizaje “AprendeU”.

En caso de no lograr desarrollar los programas y sus funciones de manera completa según lo solicitado, se calificará según lo establecido en la rúbrica.

Consideraciones y restricciones:

1. Para el manejo de datos de este proyecto debe utilizar el tipo de datos “lista” de la clase list(), mediante las técnicas de manipulación de listas.
2. No se permite el uso de diccionarios (dict), conjuntos (set) o tuplas (tuple) de lo contrario tendrá un cero en la evaluación.
3. Como se indicó anteriormente, para el manejo de datos solo se permite el uso de listas, pero los siguientes métodos de listas no están permitidos utilizarlos en el proyecto, de lo contrario tendrá un cero en la evaluación: insert(), pop(), remove(), reverse(), sort(), count(), index() y extend(). Tampoco está permitido el uso de la función “del”.
4. En caso de detectarse código fuente autogenerado por herramientas de inteligencia artificial tal como ChatGTP, entre otros, los apartados desarrollados con estas herramientas tendrán un cero en el puntaje.
5. Para la evaluación del proyecto se realizará una sesión con el tutor para presentar los resultados del estudiante mediante la ejecución del

programa y sus diferentes funcionalidades, así como creación, modificación y ajuste del código fuente como parte de la evaluación del conocimiento del estudiante. Se enviará al estudiante una lista con horarios de disponibilidad para que pueda seleccionar la hora de la cita a convenir. Esta revisión permitirá verificar el conocimiento adquirido del estudiante y el desarrollo de habilidades de lógica.

6. Para cada funcionalidad desarrollada debe solicitar la información al usuario mediante la función "input()", controlando las funciones mediante opciones de menú.

Sistema de Seguimiento de Signos Vitales en Pacientes

El Centro Médico Vida Saludable es una clínica especializada en el seguimiento y monitoreo de la salud de sus pacientes. Dentro de sus servicios, se encuentra el control de signos vitales como presión arterial, frecuencia cardíaca y temperatura, los cuales son registrados en cada consulta médica.

Para garantizar un correcto funcionamiento del sistema, se debe desarrollar una aplicación en Python que permita registrar pacientes, doctores y sus consultas, asegurando el almacenamiento adecuado de los datos en listas.

Requerimientos Iniciales

Antes de que el sistema entre en operación, se deben cargar previamente los datos de al menos dos pacientes y dos doctores en las respectivas listas.

Ejemplo de Datos Precargados:

Pacientes:

1. Cédula: 12345678, Nombre: Ana Pérez, Teléfono: 8888-8888, Correo: ana@mail.com
2. Cédula: 87654321, Nombre: Carlos Gómez, Teléfono: 7777-7777, Correo: carlos@mail.com

Doctores:

1. Cédula: 55566677, Nombre: Dra. María Ruiz, Especialidad: Cardiología
2. Cédula: 44455566, Nombre: Dr. Juan Rojas, Especialidad: Pediatría

Estos datos deben ser incluidos en el código antes de la ejecución del sistema, de manera que se pueda realizar la gestión de consultas y signos vitales sin necesidad de ingresar nuevos registros manualmente en la primera ejecución.

Funciones del Sistema

El sistema deberá permitir:

- Registrar nuevos pacientes y doctores.
- Registrar signos vitales de los pacientes en consultas médicas.
- Validar que los doctores y pacientes existan antes de asociar registros.
- Generar reportes de pacientes con hipertensión.
- Calcular el promedio de signos vitales de un paciente.
- Buscar consultas por fecha específica.

Menú de Opciones

1. Registrar Paciente
2. Registrar Doctor
3. Registrar Control de Signos Vitales
4. Ver Historial de Signos Vitales de un Paciente
5. Generar Reporte de Pacientes con Hipertensión
6. Calcular Promedio de Signos Vitales de un Paciente
7. Buscar Consultas por Fecha
8. Salir

Requerimientos Funcionales

1. Registro de Pacientes

- Registrar nuevos pacientes con los siguientes datos:
 - Cédula (debe ser única)
 - Nombre
 - Teléfono
 - Correo electrónico
- Validar que la cédula no esté repetida.
- Almacenar la información en una lista de listas.

2. Registro de Doctores

- Registrar nuevos doctores con los siguientes datos:
 - Cédula (debe ser única)
 - Nombre
 - Especialidad
- Validar que la cédula no esté repetida.
- Almacenar la información en una lista de listas.

3. Registro de Controles de Signos Vitales

- Registrar un control de signos vitales asociando:
 - Paciente (identificado por su cédula).
 - Doctor (identificado por su cédula).
 - Fecha de consulta.
 - Lista de signos vitales, donde cada signo vital incluirá:
 - Nombre del signo vital (Ejemplo: "Presión Arterial", "Frecuencia Cardíaca", "Temperatura").
 - Valor medido (Ejemplo: "120/80 mmHg", "72 bpm", "36.5 °C").
- Validar que el paciente y el doctor existan.

- Usar listas anidadas para almacenar los controles de cada paciente.
- Mostrar alerta si los valores están fuera del rango normal:
 - **Presión Arterial Normal:** 90/60 - 130/80 mmHg. Si el valor es superior a 130/80 mmHg o inferior a 90/60 mmHg, mostrar una advertencia de presión alta o baja.
 - **Frecuencia Cardíaca Normal:** 60 - 100 bpm. Si el valor es menor a 60 bpm o mayor a 100 bpm, mostrar una advertencia de bradicardia o taquicardia.
 - **Temperatura Normal:** 36.0 - 37.5 °C. Si el valor es menor a 36.0 °C, indicar posible hipotermia; si es mayor a 37.5 °C, indicar posible fiebre.

4. Historial de Signos Vitales del Paciente

- Permitir consultar el historial de signos vitales de un paciente a partir de su cédula.
- Mostrar los controles en orden cronológico, con información detallada de cada consulta.

5. Reporte de Pacientes con Hipertensión

- Generar un reporte de pacientes cuyos últimos registros tengan presión arterial alta (>130/80).
- Utilizar listas anidadas para organizar la información.
- Mostrar el reporte en pantalla.

6. Calcular Promedio de Signos Vitales de un Paciente

- Calcular el promedio de los signos vitales de un paciente según su historial.
- Si no tiene registros, mostrar un mensaje indicando que no hay datos suficientes.

7. Buscar Consultas por Fecha

- Permitir al usuario buscar consultas registradas en una fecha específica.
 - Listar los pacientes atendidos ese día y sus valores de signos vitales.
-

Estructura de Listas

Pacientes

```
pacientes = [  
    ["12345678", "Ana Pérez", "8888-8888", "ana@mail.com"],  
    ["87654321", "Carlos Gómez", "7777-7777", "carlos@mail.com"]  
]
```

Doctores

```
doctores = [  
    ["55566677", "Dra. María Ruiz", "Cardiología"],  
    ["44455566", "Dr. Juan Rojas", "Pediatría"]  
]
```

Signos Vitales

```
signos_vitales = [  
    ["12345678", [  
        ["2025-02-05", "55566677", [  
            ["Presión Arterial", "120/80 mmHg"],  
            ["Frecuencia Cardíaca", "72 bpm"],  
            ["Temperatura", "36.5 °C"]  
        ]],  
        ["2025-02-06", "55566677", [  
            ["Presión Arterial", "135/85 mmHg"],  
            ["Frecuencia Cardíaca", "75 bpm"],  
            ["Temperatura", "36.8 °C"]  
        ]]  
    ]]  
]
```

Nota importante:

Debe documentar el código fuente mediante la opción o carácter denominado “#” para incluir comentarios.

Debe incluir el documento de Word (capturas o pantallazos) de los resultados de la ejecución del programa que usted ha desarrollado, así como el archivo del código fuente “.py”. En caso de no entregar alguno de estos apartados, perderá la totalidad de los puntos de la asignación.

5. Evaluación del conocimiento:

1. La persona estudiante debe elegir una de las fechas disponibles en tiempo y forma, esto aplica únicamente para los estudiantes que han entregado los 2 archivos del proyecto de forma completa, en caso de que no elija ninguna fecha, se calificará con un cero, al no haber demostración de conocimientos.
2. Se enviará una única convocatoria por grupo mediante su correo institucional UNED, la sesión estará configurada con sala de espera, por lo tanto, la persona estudiante entrará a la sala de espera en Teams y el tutor le autoriza el ingreso cuando así corresponda, si la persona estudiante no se presenta en 5 minutos posteriores a la cita, se calificará con cero.
3. La persona debe identificarse antes de comenzar la evaluación de conocimientos, por lo tanto, debe mostrar ante la cámara la cédula de identidad.
4. La persona estudiante debe estar visible en cámara en todo momento durante la evaluación de conocimiento, donde ejecutará las funciones que se le indique, responderá a los planteamientos o cuestionamientos del tutor evaluador, es decir, debe demostrar que sabe cómo desarrollar las funciones, para qué sirven, como llegaron al resultado, o bien, se le podrá solicitar la inclusión, modificación o arreglo del código fuente.
5. A los estudiantes se les brindará 45 minutos de tiempo para la evaluación de conocimiento.
6. En caso de que exista una interrupción del servicio de electricidad o internet al momento de su turno, la persona estudiante, podrá presentar la certificación del proveedor en los próximos 3 días posteriores a la fecha de evaluación de conocimientos. En caso de no presentarse la certificación, se calificará con cero.
7. La persona estudiante debe conectarse 5 minutos antes a la sesión y debe haber realizado las pruebas correspondientes de su equipo de cómputo, así como contar con los audífonos, cámara y micrófono para la evaluación de conocimientos.
8. La persona estudiante no se puede conectar por medio de dispositivos móviles o Tablet, en caso de ser así se calificará con cero.

9. Se le informa a la persona estudiante que la sesión será grabada, por lo que debe respetar las normas de netiqueta.

Aviso Importante

Fraude o plagio académico se entiende todo acto de copiar, compartir o comparar por diversos medios como WhatsApp, respuestas de cualquier instrumento de evaluación, comprar trabajos a terceros o que terceros realicen trabajos en su nombre, parafrasear el trabajo o las ideas de otras personas en un trabajo académico sin un reconocimiento explícito de su autoría o cualquiera de las 12 modalidades de trabajo poco original que se detallan en la Unidad 0 de cada asignatura.

En los casos en los que la persona docente detecte que un participante comete plagio o fraude académico o cualquier otra anomalía académico-administrativa, procederá a tomar las medidas cautelares con el trabajo de la persona participante, asignando una calificación de cero, se informará a la Cátedra, se reportará a los involucrados y se seguirá el debido proceso administrativo de acuerdo con lo estipulado en el Reglamento General Estudiantil. En caso de demostrar que son inocentes de los cargos, se recalificará la actividad según corresponda.

Rúbrica de calificación

Criterios	Excelente (3pts)	Muy bueno (2pts)	Necesita Mejorar (1pts)	No se evidencia (0pts)
Formato, portada, tabla de contenido y ortografía.	Se evidencia que cumple los puntos solicitados de formato, portada, tabla de contenido y ortografía.	Se evidencia que cumple 3 de los puntos solicitados.	Se evidencia que cumple entre 2 y 1 de los puntos solicitados.	No se evidencia la información solicitada, no realiza la entrega del instrumento o lo deja en bandeja de borrador.
Funciones Python	<p>Se evidencia que desarrolla entre 7 a 6 funciones solicitados de forma completa y correcta, cumpliendo los siguientes procedimientos:</p> <p>Procedimientos:</p> <ul style="list-style-type: none"> • Crea función, define variables y valores de entrada de datos. • Realiza operaciones condicionales para el cálculo solicitado de manera correcta. • Imprime o muestra correctamente en pantalla la solución al problema descrito, según enunciado. • Incluye la documentación interna del código fuente desarrollado para la solución. • Incluye documentación técnica (capturas o pantallazos) de los resultados de la ejecución del programa desarrollado. 	Se evidencia que desarrolla entre 5 a 4 funciones solicitados de forma completa y correcta, cumpliendo con los procedimientos mencionados anteriormente.	Se evidencia que desarrolla entre 3 a 1 funciones solicitada de forma completa y correcta, cumpliendo con los procedimientos mencionados anteriormente.	No se evidencia la información solicitada, no realiza la entrega del instrumento, lo deja en bandeja de borrador o no entrega el archivo Python extensión .py ni los resultados de ejecución en Word (capturas de pantalla) o ambos.
Evaluación del conocimiento	Se evidencia que la persona estudiante integra los tres elementos de forma correcta: Crea función, define variables y valores de entrada de datos.	Se evidencia que la persona estudiante integra al menos dos elementos de forma correcta.	Se evidencia que la persona estudiante integra al menos un elemento de forma correcta.	No se presenta o no se evidencia ningún elemento de forma correcta.
	Se evidencia que la persona estudiante demuestra el conocimiento del trabajo desarrollado de al menos 3 funciones de forma correcta.	Se evidencia que la persona estudiante demuestra el conocimiento del trabajo desarrollado de al menos 2 función de forma correcta.	Se evidencia que la persona estudiante demuestra el conocimiento del trabajo desarrollado de al menos 1 función de forma correcta.	No se presenta o no se evidencia el conocimiento de la estructura del trabajo de forma correcta.

Criterios	Excelente (3pts)	Muy bueno (2pts)	Necesita Mejorar (1pts)	No se evidencia (0pts)
	Se evidencia que la persona estudiante realiza al menos tres ajustes en el código fuente (mejora, corrección o inclusión) solicitados por el tutor de manera completa y correcta.	Se evidencia que la persona estudiante realiza al menos dos ajustes en el código fuente (mejora, corrección o inclusión) solicitados por el tutor de manera completa y correcta.	Se evidencia que la persona estudiante realiza al menos un ajuste en el código fuente (mejora, corrección o inclusión) solicitados por el tutor de manera completa y correcta.	No se presenta o no se evidencia el conocimiento de la estructura del trabajo de forma correcta.
Total	15 pts	Pts obtenidos		Nota