|  |  |  |
| --- | --- | --- |
|  | **FICHA TÉCNICA DE CONSTRUCCIÓN DEL ÍTEM** | |
|  | **No. Ítem**: **1** |
|
|
| **DATOS DEL ÍTEM** | | **DATOS DEL AUTOR** |
| **Programa académico**: Ingeniería de Sistemas | |  |
| **Prueba**: Algoritmos y Programación | |  |
|  |
|  |
| **ÍTEM: COMPETENCIA ESPECÍFICA, CONTEXTO, ENUNCIADO Y OPCIONES DE RESPUESTA** | | |
| **Competencia específica señalada en el syllabus, que evalúa este ítem:**  Elabora algoritmos como estrategia de solución a problemas propuestos usando adecuadamente las estructuras de control, funciones y arreglos. | | |
| **CONTEXTO - Caso - situación problémica**:  En los lenguajes de programación, existe la sentencia de control while, que permite iterar un bloque de código mientras una condición sea verdadera. Para terminar una iteración definida por la sentencia while desde el interior del bloque de código iterado y hacer que el control del programa pase a la siguiente sentencia después del while. | | |
| **ENUNCIADO**:  ¿Cuál de las siguientes instrucciones interrumpe el flujo normal de ejecución de un ciclo while finalizándolo?: | | |
| **Opciones de respuesta**  a. Break.  b. Continue.  c. Exit.  d. End. | | |
|
|
| **JUSTIFICACIÓN DE OPCIONES DE RESPUESTA** | | |
| Por qué NO es b: porque la sentencia continue pasa el control de nuevo al ciclo while para ejecutar una nueva iteración si la condición de evaluación es verdadera. | | |
| Por qué NO es c: porque exit no es una sentencia de control del lenguaje Python. | | |
| Por qué NO es d: porque end no es una instrucción válida en lenguaje Python. | | |
| **CLAVE Y JUSTIFICACIÓN.**  La clave es a porque cuando se encuentra una sentencia break dentro de un ciclo, el ciclo se termina inmediatamente y el control del programa pasa a la sentencia siguiente al ciclo. | | |
| **ESPECIFICACIONES DE DISEÑO: DIBUJOS, ECUACIONES Y / O GRÁFICOS**: | | |