

UNIDAD DIDACTICA

TÍTULO: INTRODUCCIÓN A LA PROGRAMACIÓN.

INTRODUCCIÓN E IMPORTANCIA:

La importancia de esta unidad didáctica estriba en la familiarización del alumno con la metodología requerida para la resolución de problemas a través de medios informáticos; presentándoles los elementos básicos en la programación.

RELACIÓN CON OTRAS UNIDADES:

Esta unidad es el punto de partida para la correcta asimilación de las posteriores unidades didácticas del módulo.

TEMPORIZACIÓN:

La unidad didáctica que presento será impartida en un total de 22 sesiones (horas) de actividad docente, incluyendo las actividades de: introducción, desarrollo, refuerzo, ampliación y evaluación de la misma.

CAPACIDADES TERMINALES:

- Elegir y definir una estructura de datos para resolver un problema con lenguajes estructurados.
- Aplicar la metodología de desarrollo estructurado para el diseño de algoritmos.

OBJETIVOS DIDÁCTICOS:

Identificar un problema.

- Describir los datos del problema (reconocer).
- Clasificar y comparar los diferentes datos que pueden aparecer en un problema.
- Conocer los distintos operadores, expresiones e instrucciones que se pueden utilizar a la hora de resolver un problema.
- Aplicar pseudocódigo para obtener la solución de un problema sencillo.
- Dibujar organigramas que definan el problema.

CONTENIDOS:

1. Conceptuales:

- Aprendizaje de los pasos para la solución de un problema
- Comprensión del porqué de la utilización de diferentes tipos de datos.
- Conocimiento de los tipos diferentes de datos y sus características.

- Utilización del pseudocódigo para la resolución de problemas sencillos.
- Aprendizaje del uso de operadores, expresiones e instrucciones para la implementación de problemas en pseudocódigo.
- Utilización del organigrama como técnica alternativa de resolución de problemas.

2. Procedimentales:

- Confeccionar una estrategia mental para abordar un problema.
- Componer una serie de pasos que lleven a la solución de problemas.
- Aplicar pseudocódigo para obtener la meta fijada.
- Probar diferentes técnicas de solución de problemas.
- Utilizar los tipos adecuados de datos para un planteamiento dado.
- Manejar con soltura operadores, expresiones e instrucciones.
- Simular el proceso de solución del problema con el fin de encontrar errores de planteamiento del problema.

3. Actitudinales:

- Percatarse de la importancia de los diferentes tipos de datos.
- Apreciar la importancia de un buen planteamiento del problema.
- Ser consciente de que no hay una única solución a un problema dado.
- Conocer las distintas técnicas de resolución de problemas.
- Criticar y aceptar los métodos presentados.

ACTIVIDADES DE ENSEÑANZA- APRENDIZAJE:

A. Actividades de introducción:

- Baterías de preguntas sobre el concepto que tiene el alumno de la programación.
- Lluvia de ideas sobre la programación.
- Test individual sobre nociones de programación y los elementos que la componen.

B. Actividades de desarrollo:

- Entrego una lista con una serie de problemas sencillos con la intención de que el alumno empiece a aprender como poder afrontarlos. La clase la programaría mediante un debate en el cual discutir el diferente enfoque al que puede llegar el alumno.
- De los problemas extraídos de la actividad anterior, propongo al alumno que piense que tipo de dato sería el más apropiado para su implementación y cual no. La clase se desarrollaría con la corrección de cada una de las cuestiones en la pizarra.
- Dada la misma enumeración de problemas, se les indica que los han de resolver mediante una serie de pasos usando pseudocódigo y con

los componentes de la programación (operadores, expresiones e instrucciones) que hemos aprendido. Se les dejaría un tiempo para la realización de los mismos y luego se corregirían en la pizarra.

- Buscar ayudándose de la herramienta de internet problemas resueltos mediante organigramas y observar la equivalencia de la simbología que se usa en la herramienta gráfica con la solución de problemas utilizando pseudocódigo. Esta actividad también se resolvería realizando un debate en clase.

C. Actividades de refuerzo:

- Dados unos problemas resolverlos utilizando organigramas.

D. Actividades de ampliación:

- Plasmar un problema en un algoritmo sencillo.

E. Actividades de Evaluación:

- Realización de un examen basado en los conocimientos adquiridos en este primer tema de introducción a la programación.

METODOLOGÍA:

La metodología utilizada será una mezcla de metodología expositiva y de descubrimiento o investigadora. De esta forma fomentaré la motivación en el alumnado, ya que ellos formarán parte de su propio aprendizaje, teniéndome siempre a su lado para cualquier duda o consulta.

De esta manera se rompe con el aprendizaje unidireccional, el alumno era sólo el que debía aprender, y se fomenta el aprendizaje bidireccional, en el que tanto el alumno como el profesor generan y aprenden conocimientos nuevos.

Una jornada normal (cuando digo normal me quiero referir a que no haya programada ninguna prueba o actividad extraordinaria) comenzaría con un breve repaso del día anterior y la resolución de dudas que pudieron surgir en la última clase, también se corregirían ejercicios (si hubiera alguno pendiente para su corrección). Inmediatamente después se empezaría a impartir los contenidos que tengo programados para el presente día. Terminaría la clase con la resolución de actividades por parte del alumno relacionadas con el contenido que se ha impartido. El tiempo que dedicaría a cada una de las partes en las que he dividido la clase dependerían de si el horario que posee el modulo en ese día consta de una, dos o incluso tres sesiones (considerando las sesiones de 1 hora).

CRITERIOS DE EVALUACIÓN:

- Citar operaciones que permiten realizar una estructura de datos desde un programa y explicar sus algoritmos.
- Justificar la importancia de la adecuada selección de estructuras de datos para la resolución de problemas en programación.
- Sobre un problema de programación en gestión propuesto:
 - Elegir las estructuras más adecuadas para representar y manejar los datos del problema
 - Describir los algoritmos de tratamiento de las estructuras para la resolución del problema.
- Identificar las estructuras básicas de programación.

EVALUACIÓN:

La evaluación es un proceso continuo y flexible y puede dividirse en varios momentos:

- La evaluación inicial o de diagnóstico, será de la que se extraigan los conocimientos básicos que tiene el alumno sobre el tema de programación.
- Evaluación continua o formativa, es aquella en la que tomando como instrumentos la observación directa y pequeñas pruebas sencillas, se valorará como el alumno avanza desde sus conocimientos iniciales hasta los objetivos anteriormente mencionados. Se valorará como punto muy importante la participación y el esfuerzo.
- Evaluación final o sumativa, incluye a los dos momentos anteriores y en ella valoraré como el alumno ha adquirido los objetivos y contenidos propuestos para esta unidad. Esta valoración la obtendré a través de una prueba individual escrita.

Calificación numérica:

La valoración de los conocimientos se llevará a cabo mediante calificación numérica, concretándose en valores de 1 a 10 puntos. De forma orientativa, se pueden fijar los siguientes valores: conceptos (40%), procedimientos (40%) y actitudes (20%). El alumno superará la prueba si obtiene un valor superior o igual al 5.

RECURSOS

Los recursos que utilizaré para poder desarrollar esta unidad de trabajo serán los siguientes:

- Pizarra.
- Retroproyector y portátil.
- Fotocopias.
- Ordenadores personales e Internet.

TEMAS TRANSVERSALES

En el desarrollo del proceso de enseñanza-aprendizaje que llevo acabo en esta unidad lleva implícito la coeducación y el trabajo en grupo.

También, al considerar que tengo alumnos de otras nacionalidades y religiones, fomento la educación para la convivencia y la paz.