**Tarea 4: Repaso Unidad + Lenguajes de programación**

**Desarrollo**

1. Indica la función principal de los siguientes softwares de programación:
   1. Editores de código o texto
   2. Compiladores
   3. Depuradores
   4. Enlazadores
   5. Interpretadores o traductores
2. Completa la siguiente tabla comparativa entre Java, C++ y Python.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Java | C++ | Python |
| Utilizado para crear |  |  |  |
| Pros |  |  |  |
| Contras |  |  |  |

1. Indica las características principales y los lenguajes soportados de los siguientes entornos de programación de escritorio:
   1. Eclipse
   2. NetBeans
   3. Visual Studio
   4. VS Code
2. Busca e indica 2 editores de código online e indica qué lenguajes soporta cada uno.
3. Indica los criterios o factores que se deben de tener en cuenta a la hora de seleccionar un lenguaje de programación.
4. Indica qué lenguaje(s) seleccionarías para cada opción, e indica el motivo:
   1. Móvil
   2. Aplicaciones de escritorio
   3. Ciencias
   4. Desarrollo web
      1. Frontend
      2. Backend
5. ¿Podríamos programar sin lenguaje de programación? Argumenta tu respuesta.

**Test**

1. Tipo de lenguajes en el que el diseño de los programas se basa más en los datos y su estructura; es decir, que la unidad de proceso es el objeto y en él se incluyen los datos (variables) y operaciones que actúan sobre ellos:
   1. Declarativos
   2. Imperativos
   3. Orientados a objetos
2. Sistema de códigos interpretable directamente por un circuito microprogramable que se compone de un conjunto de instrucciones que determinan acciones que serán realizadas por la máquina:
   1. Lenguaje de bajo nivel
   2. Lenguaje máquina
   3. Lenguaje de alto nivel
3. Un intérprete traduce los símbolos de un lenguaje de programación a su equivalente escrito en lenguaje máquina (proceso conocido como compilar); por último, se obtiene un programa ejecutable:
   1. Verdadero
   2. Falso
4. Lenguaje de programación que proporciona poca capacidad de servicio o funcionamiento del microprocesador de una computadora:
   1. Lenguaje de bajo nivel
   2. Lenguaje máquina
   3. Lenguaje de alto nivel
5. Herramienta que permite desarrollar software o programas cuyo objetivo principal es el de la solución de problemas en una computadora:
   1. Lenguaje de programación
   2. Diseño web
   3. Base de datos
6. Lenguaje que se caracteriza porque su estructura semántica es muy similar a la forma como escriben los humanos, lo que permite codificar los algoritmos de manera más natural, en lugar de codificarlos en el lenguaje binario de las máquinas o a nivel de lenguaje ensamblador:
   1. Lenguaje de bajo nivel
   2. Lenguaje de alto nivel
   3. Lenguaje máquina
7. Programa que analiza la información, la ejecuta en un código fuente, toma y procesa un código, lo traduce y lo ejecuta:
   1. Compilador
   2. Intérprete
   3. Sintaxis
8. Las fases de la compilación son:
   1. Creación de código fuente y compilación del programa
   2. Compilación del programa
   3. Creación de código fuente y enlace del programa con las funciones necesarias de la biblioteca.
   4. Todas las opciones anteriores son incorrectas.
9. ¿Cómo se llama el proceso de traducción de código fuente a código objeto?
   1. Interpretación
   2. Ejecución
   3. Compilación
10. Si queremos construir una aplicación pequeña, y se prevé que no sufrirá grandes cambios durante su vida, ¿sería el modelo de ciclo de vida en cascada el más recomendable?
    1. No
    2. Sí
11. El compilador recibe código fuente y genera:
    1. Código máquina
    2. Código objeto
    3. Ninguna de las anteriores
12. Relaciona los tipos de códigos con su característica más relevante:
    1. Código ejecutable
    2. Código objeto
    3. Código fuente
    4. Escrito en lenguaje máquina pero no ejecutable
    5. Escrito en algún lenguaje de programación de alto nivel, pero no ejecutable
    6. Escrito en lenguaje máquina y directamente ejecutable.
13. ¿En qué modelos el resultado de una fase es la entrada de la siguiente?
    1. Modelo en V
    2. Modelo en cascada
    3. Todos los anteriores
14. ¿Cuál es la fase más importante del desarrollo de software?
    1. Análisis
    2. Codificación
    3. Diseño
15. Indica la opción falsa sobre el modelo en cascada:
    1. Es un modelo bastante rígido
    2. Las fases pueden realizarse sin seguir un orden.
    3. Identifica las fases principales del desarrollo de software.
16. Indica la opción correcta sobre Las metodologías ágiles:
    1. Tienen pocos roles y artefactos
    2. Grupos pequeños
    3. El cliente es parte del equipo
    4. Todas son verdaderas.
17. ¿Cuál de las siguientes opciones sobre SCRUM es falsa?
    1. Desarrollo iterativo
    2. Ciclos largos de trabajo.
    3. Flexibilidad
18. El rol de Scrum Master no se encarga de…
    1. Facilitar el proceso Scrum
    2. Resolver impedimentos
    3. Definir objetivos del proyecto