

# Activitats

## Introducció

1. Descriu breument la relació que hi ha entre els components de maquinari principals d'un ordinador i l'emmagatzematge i l'execució del programari. Pots basar-te en la màquina de Von Newman.

Algunos de los componentes tienen mucha relación con el almacenamiento y la ejecución de programas. Principalmente el disco duro, componente en el que se almacena toda la información del ordenador, incluidos los programas y además la RAM, que es la memoria encargada de los procesos de los programas, de esa forma no hay que acceder al disco duro cada vez que se quiere hacer algo en un programa que ya está ejecutándose.

2. Defineix els conceptes següents:

- Codi font.  
Es un archivo o conjunto de archivos que contienen instrucciones concretas en un lenguaje de programación para después compilar uno o más programas.
- Codi objecte.  
Es un conjunto de instrucciones y datos escritos en un lenguaje que entiende el ordenador directamente. Binario, código máquina...
- Codi executable.  
Es un lenguaje creado por la máquina que se corresponde a los programas, el ordenador realiza las instrucciones compiladas por medio de librerías y el compilador.

## Cicle de vida del programari

1. Defineix "Cicle de vida del programari".

Es el ciclo que cualquier software realiza, va desde las primeras partes de su desarrollo hasta su utilización, actualizaciones y mantenimiento...

2. Anomena les fases principals del desenvolupament de programari i explica breument que es fa a cadascuna.

La primera fase es analizar los requerimientos, en esta se tiene que ver bien que necesidades tiene que cubrir el programa, si tiene que ser más completo o más simple...

La segunda es el diseño, básicamente consiste en hacer un diseño inicial aún sin programar de como podría funcionar el programa para tener una estructura clara a la hora de empezar a programar en la siguiente fase.

La tercera es el desarrollo, aquí ya guiándose por el diseño inicial se empieza a desarrollar el programa.

La cuarta consiste en hacer pruebas, una vez desarrollado queda probarlo para comprobar que realmente cumple todas las necesidades que se pedían inicialmente y que además funciona bien.

La quinta sería la publicación, el programa ya está terminado así que se publica ya sea algo público o para un cliente en concreto...

La sexta y última fase es el mantenimiento, cualquier programa necesita cierto mantenimiento por si hay algún error y además, si surgen nuevas necesidades de aquí se puede pasar de nuevo a la primera fase.

### 3. Avantatges i inconvenients del model en cascada.

Una de las ventajas más importantes es que una vez empieza una de las fases cuenta con toda la información y todo lo conseguido en la anterior, por ejemplo el desarrollo es mucho más sencillo teniendo un diseño claro.

En cuanto a inconvenientes principalmente que puede no ser tan rápido como otros ya que para iniciar una fase hay que esperar a que finalice la anterior, aún así es el modelo que más se utiliza.

### 4. Explica com funciona el model de desenvolupament mitjançant la creació de prototips.

Basicamente consiste en ir desarrollando distintos prototipos e ir presentandoselos al cliente, es importante que el cliente tenga en cuenta que no se trata del producto final a la hora de juzgar el prototipo. Gracias a eso se consigue también ver como de viable y rentable es el producto final, además todo o como mínimo partes de prototipo se reutilizan para el proyecto final.

### 5. Quins quatre principis regeixen el desenvolupament àgil expressats al Manifest Àgil?

Individuos e interacciones por encima de procesos y herramientas.

Software funcionando por encima de documentación exhaustiva.

Colaboración con el cliente por encima de negociación contractual.

Respuesta ante el cambio por encima de seguir un plan.

### 6. Què és una història d'usuari? Consulta el següent enllaç i posa un exemple propi.

Les podem formular de la següent manera: “Com a [perfil], vull [objectiu del software], per a poder [resultat]”.

[https://es.wikipedia.org/wiki/Historias\\_de\\_usuario](https://es.wikipedia.org/wiki/Historias_de_usuario)

Es una frase corta utilizada en el desarrollo ágil para especificar algunos requisitos.

Un ejemplo podría ser: “Como cliente quiero poder utilizar mi tarjeta de crédito para comprar.”

### 7. KANBAN. Estudia els avantatges i els inconvenients de tenir una pissarra web digital per a la metodologia Kanban. Pots consultar els següents enllaços:

<https://leankit.com/learn/kanban/kanban-board/>  
<https://trello.com/es>  
<https://taiga.io/>  
<https://kanbantool.com/es/>

12. KANBAN. Fes un resum de la metodologia Kanban i indica les seves diferències davant de SCRUM. Pots consultar el següent enllaç:

<https://ca.atlassian.com/agile/kanban>

Se trata de un método visual de gestión de proyectos, con el se puede visualizar bien el flujo y carga de trabajo en un tablero Kanban. El trabajo se muestra en forma de tablero organizado por columnas.

La principal diferencia es que mientras kanban se centra en una metodología continua y fluida, scrum se basa más en trabajos cortos y estructurados.

13. SCRUM. Explica com funciona Scrum. Consulta els enllaços següents:

<https://proyectosagiles.org/que-es-scrum/>  
<https://proyectosagiles.org/como-funciona-scrum/>

En Scrum se van realizando entregas regulares y parciales del trabajo final, de manera prioritaria y en función del beneficio que aportan dichas entregas. Por este motivo, es una metodología especialmente indicada para proyectos complejos o que tengan requisitos cambiantes.

14. SCRUM. Defineix els termes següents:

- Product backlog.  
El product backlog (o pila de producto) es un listado de todas las tareas que se pretenden hacer durante el desarrollo de un proyecto.
  - Sprint backlog.  
Es básicamente una lista de tareas indentificadas por el equipo, esta lista tendrá que ser completada durante cada uno de los sprints.

15. SCRUM. A la terminologia Scrum quins termes s'utilitzen com a sinònim de:

- Cap de projecte.  
Scrum master
  - Client.  
Product Owner
  - Equip de desenvolupament.  
Squad

16. XP. Quines són les característiques distintives de XP davant d'altres metodologies àgils? Explica-les. Pots consultar el següent enllaç:

<http://www.davidvalverde.com/blog/introduccion-a-la-programacion-extrema-xp/>

En la XP se divide el proyecto en fases y cada una de ellas tiene un ciclo completo de planificación, diseño, desarrollo y pruebas. Ya que tiene una retroalimentación continua con el cliente y el equipo de desarrollo se puede utilizar para proyectos que no tengan requisitos muy precisos y que puedan cambiar.

# Llenguatges de programació

1. Quina diferència hi ha entre els llenguatges declaratius i els imperatius? Anomena almenys 2 de cada tipus.

Los imperativos (Java, Python) siguen una secuencia de operaciones que tienen que realizar.

Los declarativos (Prolog, Lisp) especifican el resultado deseado pero no como lograrlo.

2. Explica què és compilar? Explica què és interpretar?

Compilar es traducir con un compilador un programa de alto nivel a lenguaje maquina. Interpretar por otro lado es directamente ejecutar las instrucciones dadas.

3. Avantatges dels llenguatges compilats.

Una vez se genera el programa es muy fácil de distribuir ya que se puede ejecutar por si solo, no necesita un intérprete para poder ejecutarse. Por otra parte cada sistema operativo necesita una versión compilada distinta.

4. Avantatges dels llenguatges interpretats.

A pesar de necesitar el intérprete, no necesitas una versión distinta para cada sistema operativo por lo que es más fácil de distribuir a distintas plataformas.

5. Anomena 2 llenguatges compilats i altres 2 interpretats.

Compilados: C++, C

Interpretados: Java, Python

6. Es pot considerar codi objecte el **bytecode** generat a Java després de la compilació? Explica'n la resposta.

Si, ya que el bytecode es un lenguaje intermedio que se genera al compilar el programa, por lo tanto debería ser considerado código objeto.

7. Posa un exemple de llenguatge dels tipus següents:

- Baix nivell.  
Código binario
  - Nivell mitjà.  
C
  - Alt nivell.  
Java

8. Quin paradigma de programació segueixen els llenguatges següents?

- C  
Programación estructurada
  - C++  
Programacion genérica y orientada a objetos, de procedimiento

- SQL  
Multiparadigma
- Java  
Programación orientada a objetos y basado en clases
- Javascript  
Declarativo e imperativo
- Lisp  
Multiparadigma funcional por procedimientos, reflexión y metaprogramación
- Prolog  
lógico

Pots consultar el següent enllaç:

- [https://ca.wikipedia.org/wiki/Paradigma\\_de\\_programaci%C3%B3n](https://ca.wikipedia.org/wiki/Paradigma_de_programaci%C3%B3n)

9. Explica quins criteris es poden seguir a l'hora d'escollir un llenguatge de programació per al desenvolupament de programari.

El tipo de aplicación que es, si está dirigida a varias plataformas y a que plataformas, el mantenimiento que va a tener, la seguridad necesaria, el límite de tiempo que hay para el desarrollo...

### Activitat final (no realitzar encara)

Elabora les respostes de les preguntes d'aquesta Unitat i guarda l'arxiu a GitHub en un repositori anomenat 'Activitats\_ED'.