	Examen Entornos de Desarrollo Tema 1	Eduardo Martínez Romero	
		13/11/20	Página 1 de 9

Alumno: Eduardo Martínez Romero

Fecha: 13 de noviembre de 2020

Ejercicio 1 (1 punto)

Clasifica según:

- Tipo de tarea que realiza
- Método de distribución
- Licencia de software

Las siguientes aplicaciones. **Justifica tus respuestas**

7zip



- Programa para comprimir y descomprimir multitud de formatos de ficheros comprimidos. Ofrece su propio formato de compresión (7-zip) el cual ostenta uno de los mejores resultados en tiempos de compresión y descompresión que la competencia, además de alcanzar razones de compresión muy altas, reduciendo drásticamente el tamaño de sus ficheros.
- Se distribuye de manera gratuita desde su página oficial, <https://www.7-zip.org/>, compilado para distintas plataformas, oficialmente para Windows. También existen versiones no oficiales para una gran variedad de SO's como GNU/Linux, Macintosh, FreeBSD, Solaris y Amiga, entre otros.
- Se distribuye bajo la licencia LGPL (GNU Lesser Public Licence), una variante de la licencia GPL, por lo que estamos ante un **Software Libre**. Significa que tanto los binarios ejecutables del software como su código fuente se distribuyen y están a disposición de todo el que lo quiera, para usarlo, compartirlo, estudiarlo, modificarlo y redistribuirlo incluso cobrando por él. Claro que esa redistribución también deberá cumplir estas características. Dicho de otro modo, puedes vender tu modificación y cobrar por ella cuánto veas razonable, pero debes distribuir el código fuente con ella, además de acogerte a las **cuatro libertades** del software libre:

Libertad 0.- Permitir usar el programa de cualquier modo, y con cualquier propósito.

Libertad 1.- Libertad de estudiar como funciona el programa y que el usuario pueda adaptarlo a sus necesidades.

Libertad 2.- Libertad de compartir tantas copias como el usuario desee.

Libertad 3.- Libertad de mejorar el programa y hacer públicas las mejoras a los demás, ya sea de forma gratuita o cobrando por él, pero claro, siempre, de nuevo, bajo estas cuatro libertades. Esto es debido a que tanto las licencias LGPL como GPL son licencias **copyleft**. Esto significa que cualquier modificación de los programas bajo esta licencia deben de forma obligatoria seguir bajo licencias de software libre compatibles. Estas son todas las que satisfagan las cuatro libertades anteriormente mencionadas.

 	Examen Entornos de Desarrollo Tema 1	Eduardo Martínez Romero
	13/11/20	Página 2 de 9

Ejercicio 1 (Continuación)

Visual Code

- Visual Studio Code, es un editor de código, un tipo de programa diseñado para facilitar la tarea de programar y desarrollar software. Dependiendo de lo completas que acaban siendo estas herramientas en lo que a utilidades para programar y depurar código se refiere, existe otra categoría de programas llamados IDE's (Integrated Development Environment), mucho más completos y avanzados para el trabajo, tanto en solitario como en equipo. Estos editores de código pueden estar preparados para trabajar con multitud de lenguajes de programación y de marcas, aunque algunos de éstos se especialicen y de soporte sólo a algún lenguaje en concreto. En este caso, Visual Studio Code se trata de un editor de código multilenguaje, con resaltado de sintaxis, herramientas de depuración y auto completado de código. Dependiendo del lenguaje en cuestión tendrá disponible una o varias de estas características.
- Este programa se distribuye oficialmente de manera gratuita desde <https://code.visualstudio.com/>, con versiones para MS Windows, distintos paquetes para Linux y MacOS, pero estas implementaciones oficiales son privativas, debido a que Microsoft cambia la licencia de este programa en sus implementaciones, probablemente debido a que existen funcionalidades y porciones de código que no quiere liberar, por lo que sólo ofrece los binarios de estas implementaciones pero no su código fuente. Sin embargo, el proyecto base sí está liberado y disponible para su estudio, implementación y modificación, entre otras posibles acciones.
- Su código fuente se distribuye bajo la licencia MIT (Massachusetts Institute of Technology). Esta licencia compatible con la licencia GPL es una de las más utilizadas actualmente. Ofrece unas características muy similares a otras licencias de software libre, aunque en este caso no es **copyleft**, por lo que los desarrolladores que realicen modificaciones de este software pueden cambiar su licencia por otras más privativas, perdiendo en ese caso la denominación de software libre.

Ejercicio 2 (2 punto)

El centro IES Doñana nos ha pedido que realicemos una aplicación para que podamos controlar las notas de los alumnos y su comportamiento en clase, para lo cual nos ha dado una serie de requisitos. Durante el desarrollo de la aplicación se nos han dado los siguientes casos, identifica en que etapa del ciclo de vida software nos encontrábamos en cada momento. **Justifica tus respuestas. En caso de que exista fase de documentación o mantenimiento justificar también el tipo:**

- Se ha desarrollado una funcionalidad que permite enviar un correo electrónico al alumno, pero se necesita comprobar si el correo llega o no llega.

Se trata de un proceso del periodo de pruebas; etapa del desarrollo que se fundamenta en verificar y asegurar que el programa realiza las funciones para las que se ha realizado, y de este modo poder satisfacer las necesidades del cliente.

- Tras leer los requisitos, le decimos a nuestro cliente para que nos valide qué para hacer el cálculo de la nota media trimestral de los alumnos, el usuario deberá introducir todas las notas de los exámenes de un alumno. Cuando el usuario pulse en calcular nos devolverá la nota media.



Nos encontramos en la **etapa de análisis**, discutiendo los requisitos que deberá cumplir nuestro programa, y por ello tratamos de confirmar con nuestro cliente detalles sobre cómo nuestro programa cumplirá determinados requisitos funcionales. En esta ocasión queremos que nos de luz verde para que el requisito funcional de calcular la nota media de los alumnos deba cumplir un requisito no funcional, pero estrechamente relacionado con esta funcionalidad. Se trata de asegurarse de que para que se realice dicho cálculo, el usuario que se encuentre utilizando el programa deberá haber introducido todas las notas de los exámenes del alumno previamente. Esta información es de gran interés para confeccionar mas tarde la **documentación de usuario**, con la que explicar a los futuros usuarios del programa como trabajar con él.

- Los programadores una vez leídos los requisitos funcionales se disponen a realizar una página que sea capaz de listar a todos los alumnos matriculados de una asignatura.

Llegados a este punto, el programa se encuentra en la **etapa de diseño**, donde los programadores tratan de maquetar el programa para que cumpla los requisitos por los que será usado. Dicho de otro modo, traducen los requisitos para poder solucionarlos, con lo que obtiene un boceto del programa, una representación parcial de cómo será y como se comportará.

- Tras un año de funcionamiento exitoso el centro ha decidido que quiere crear una nueva sección de histórico para guardar las notas de los alumnos de años anteriores

Se trata de una actividad perteneciente a la **etapa de mantenimiento**. Tras un uso satisfactorio del programa, aparecen nuevas necesidades que no fueron contempladas durante las etapas iniciales de su desarrollo, y ahora el cliente nos solicita que podamos **actualizar** el software, de manera que podamos incluir una nueva característica.

 	Examen Entornos de Desarrollo Tema 1	Eduardo Martínez Romero	
		13/11/20	Página 4 de 9

Ejercicio 2 (Continuación)

- Describimos de qué manera vamos a guardar la información en nuestro sistema, cuál va a ser nuestra arquitectura interna

Volvemos a encontrarnos en la etapa **de diseño**, tratando de ponernos de acuerdo respecto a **cómo** vamos a cumplir los requisitos que se nos solicitan. En el enunciado se ve bien claro que quieren decidir cómo guardaran la información que genere nuestro programa, como se organizará esa información, etcétera.

- Se conforma en una serie de archivos la manera en la que hay que proceder para que la aplicación que hemos creado esté funcionando y qué hacer en caso de que no esté disponible.

Estamos ante la confección de la **documentación para usuarios**, la cual forma parte de la **etapa de documentación**. En ella se le explicará, tanto a los administradores como a los usuarios, cómo utilizar el programa correctamente para obtener los resultados que se desean, al igual que cómo actuar en caso de problemas con él.

- Se coge el servidor del centro IES Doñana y se le instala la base de datos necesaria para que funcione nuestra aplicación. Una vez realizada esta tarea instalaremos la aplicación en el móvil para ver que existe una comunicación correcta.

Es una actividad de implementación, perteneciente a la **etapa de explotación**. Vamos a instalar el programa, configurarlo y realizar una prueba de funcionamiento para confirmar que todo va como debe.

Ejercicio 3 (1.5 punto)

Identifica los siguientes requisitos según sean requisitos funcionales y requisitos no funcionales:

- Los alumnos podrán ver sus notas en la plataforma
- La nota de los alumnos no podrá ser mayor a 10 ni menor que 0
- El sistema de gestión de base de datos debe ser MySQL
- El profesor permitirá guardar la nota de un alumno
- El profesor podrá cambiar la nota de un alumno
- El sistema debe ser implementado según el patrón modelo-vista-controlador (MVC)
- El sistema deberá poder informar a los alumnos automáticamente cuando tengan su nota puesta a través de correo electrónico
- El sistema debe facilitar la impresión de un listado de notas
- La interfaz del sistema debe ser una aplicación Android nativa
- El sistema debe poder usar los servicios proporcionados por Séneca para su comunicación

Requisitos Funcionales	Requisitos No Funcionales
- Los alumnos podrán ver sus notas en la plataforma	- La nota de los alumnos no podrá ser mayor a 10 ni menor que 0
- El profesor permitirá guardar la nota de un alumno	- El sistema de gestión de base de datos debe ser MySQL
- El profesor podrá cambiar la nota de un alumno	- El sistema debe ser implementado según el patrón modelo-vista-controlador (MVC)
- El sistema deberá poder informar a los alumnos automáticamente cuando tengan su nota puesta a través de correo electrónico	- La interfaz del sistema debe ser una aplicación Android nativa
- El sistema debe facilitar la impresión de un listado de notas	- El sistema debe poder usar los servicios proporcionados por Séneca para su comunicación

Ejercicio 4 (1.5 puntos)

Realiza los niveles 0,1,2 del diagrama de flujo de datos del sistema descrito en las siguientes líneas:

La aplicación que ha pedido el IES Doñana sirve para poder gestionar a los alumnos.

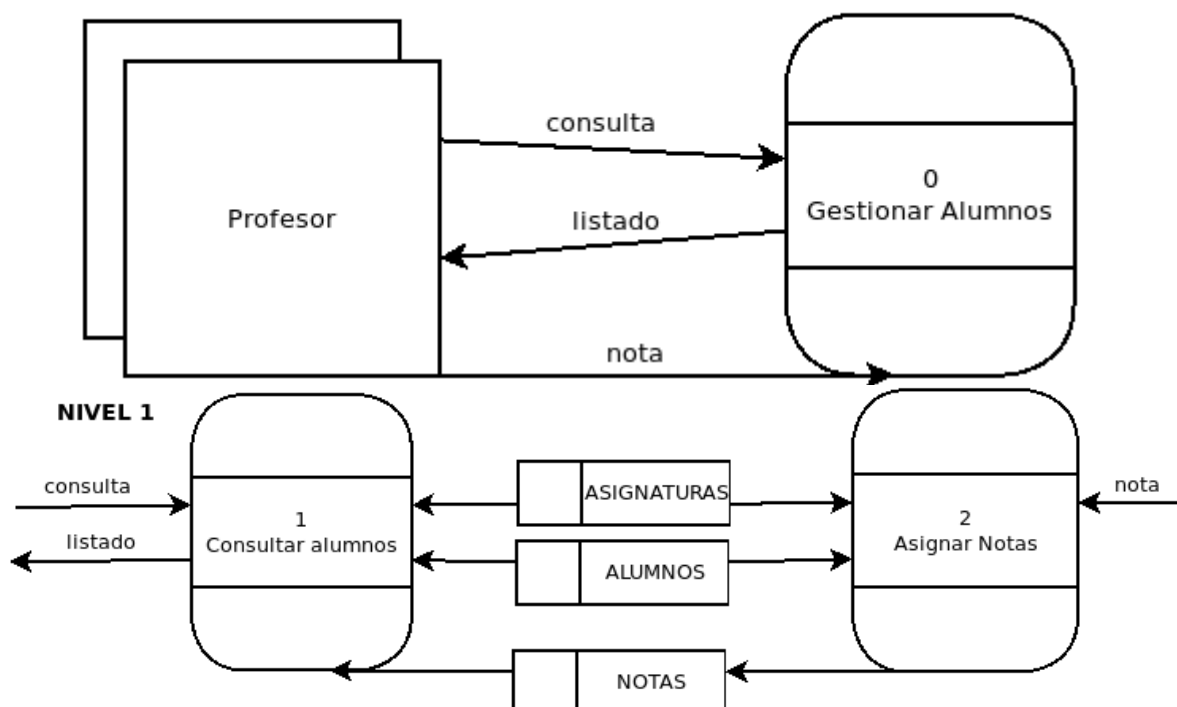
En esta aplicación, el profesor puede realizar la consulta de los alumnos que hay en una asignatura. Para ello, el profesor debe seleccionar la asignatura en el desplegable y el sistema leerá en la base de datos los alumnos que hay en la clase y los devolverá en forma de listado. En el listado aparecerán: nombre y apellidos, edad, correo electrónico y notas de los exámenes realizados si ha hecho alguno.

Por otra parte, el profesor puede seleccionar a un alumno y asignarle una nota en una de las asignaturas de las que esté matriculado el alumno. Para ello, el profesor debe seleccionar al alumno, posteriormente seleccionar la asignatura a la que quiera poner la nota, ponérsela y almacenar los cambios en el sistema. Las notas pueden ser: Menor a 5 -> suspenso, 5 a 6 -> suficiente, 6 a 7 -> bien, 7 a 9 -> Notable, 9 a 10 -> Sobresaliente.

Voy a usar la **notación de Gane y Sarson**, debido a que en la aplicación Dia no está la notación de Yourdon. A pesar de representarse de forma distinta son totalmente análogos.

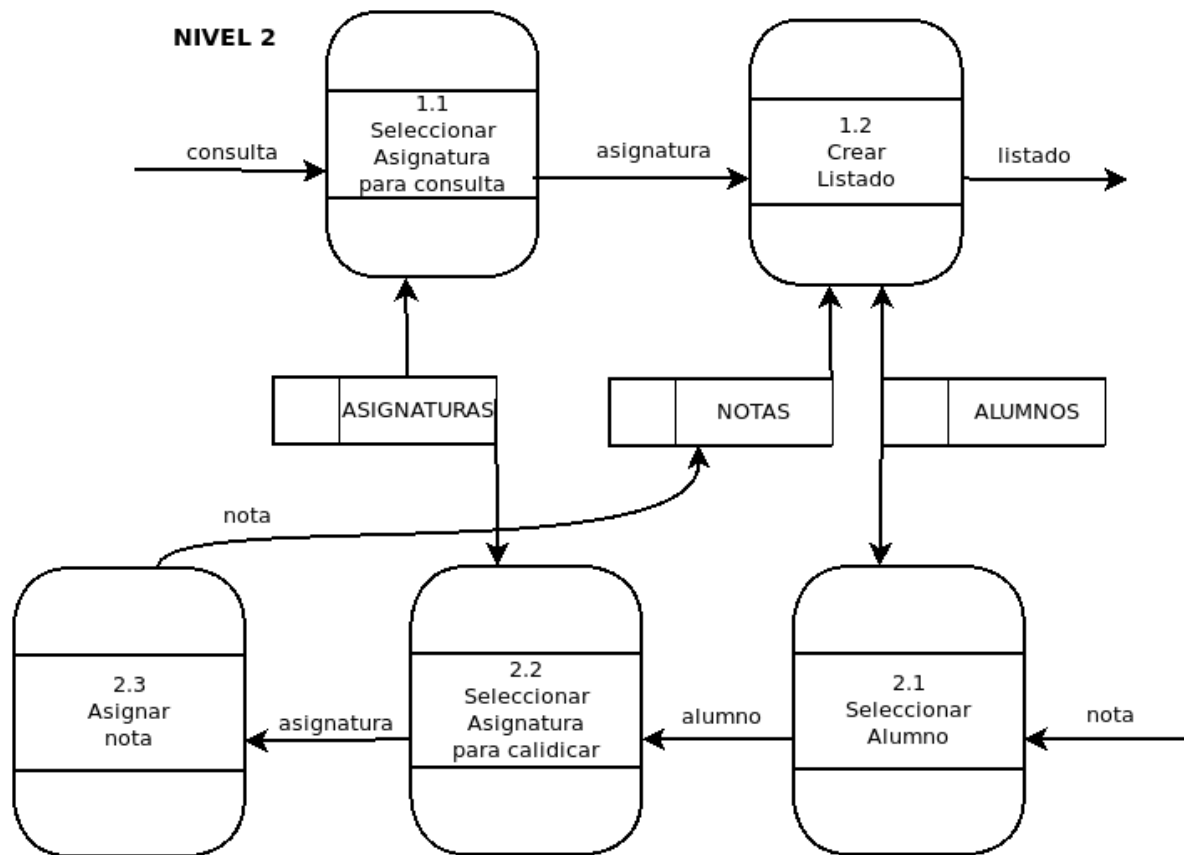
Sólo represento a las entidades externas en el nivel 0. Los almacenes de datos llevan su denominación en MAYÚSCULAS.

NIVEL 0: DIAGRAMA DE CONTEXTO





Ejercicio 4 (Continuación)



Ejercicio 5 (2 puntos)

Realiza un diagrama de flujo que:

- Muestre por pantalla los 50 primeros números que no sean el 12, 26, 44 ni 56

Pseudocódigo

INICIO

Definir num

Definir limite

Mostrar "Introduce un numero: "

Leer num

limite = num + 50

Repetir

Si num != 12 **Hacer**

Si num != 26 **Hacer**

Si num != 44 **Hacer**

Si num != 56 **Hacer**

Mostrar num

Fin si

Fin si

Fin si

Fin si

num = num + 1

Hasta que num == limite

FIN

Comentarios

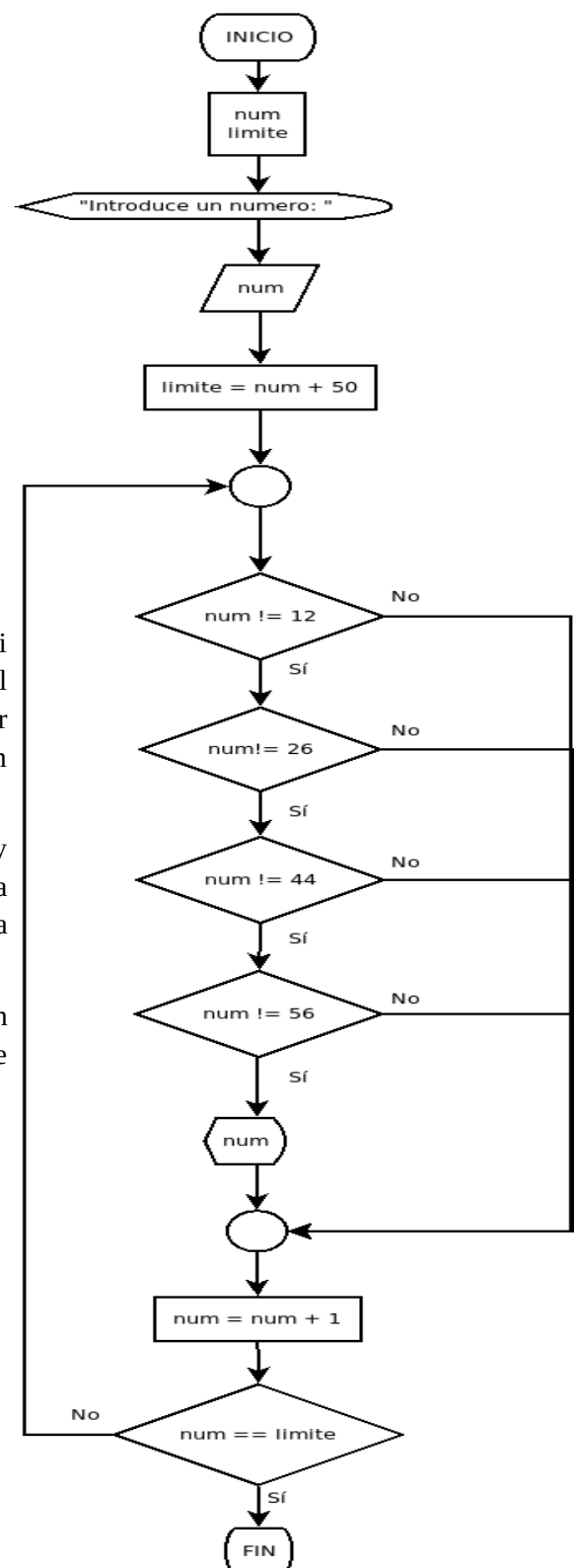
Pensé en que el enunciado podía ser ambiguo, debido a que si muestro los primeros 50 números, y contemplo no mostrar el 56, puede significar que no siempre se empiece desde cero. Por eso me decidí a plantear el ejercicio con la capacidad de leer un número, y a partir de éste mostrar sus 50 primeras sucesiones.

En el caso de que no fuera lo que el enunciado pide, estoy asegurando demostrar que los conocimientos necesarios para resolverlo han sido aplicados. Por eso añado un pseudocódigo, a pesar de no solicitarse, para poder justificar el diagrama

Como curiosidad realicé una implementación de este código en Java, pero sin recibir nada por teclado, para probar que funcionaban las estructuras de repetición.

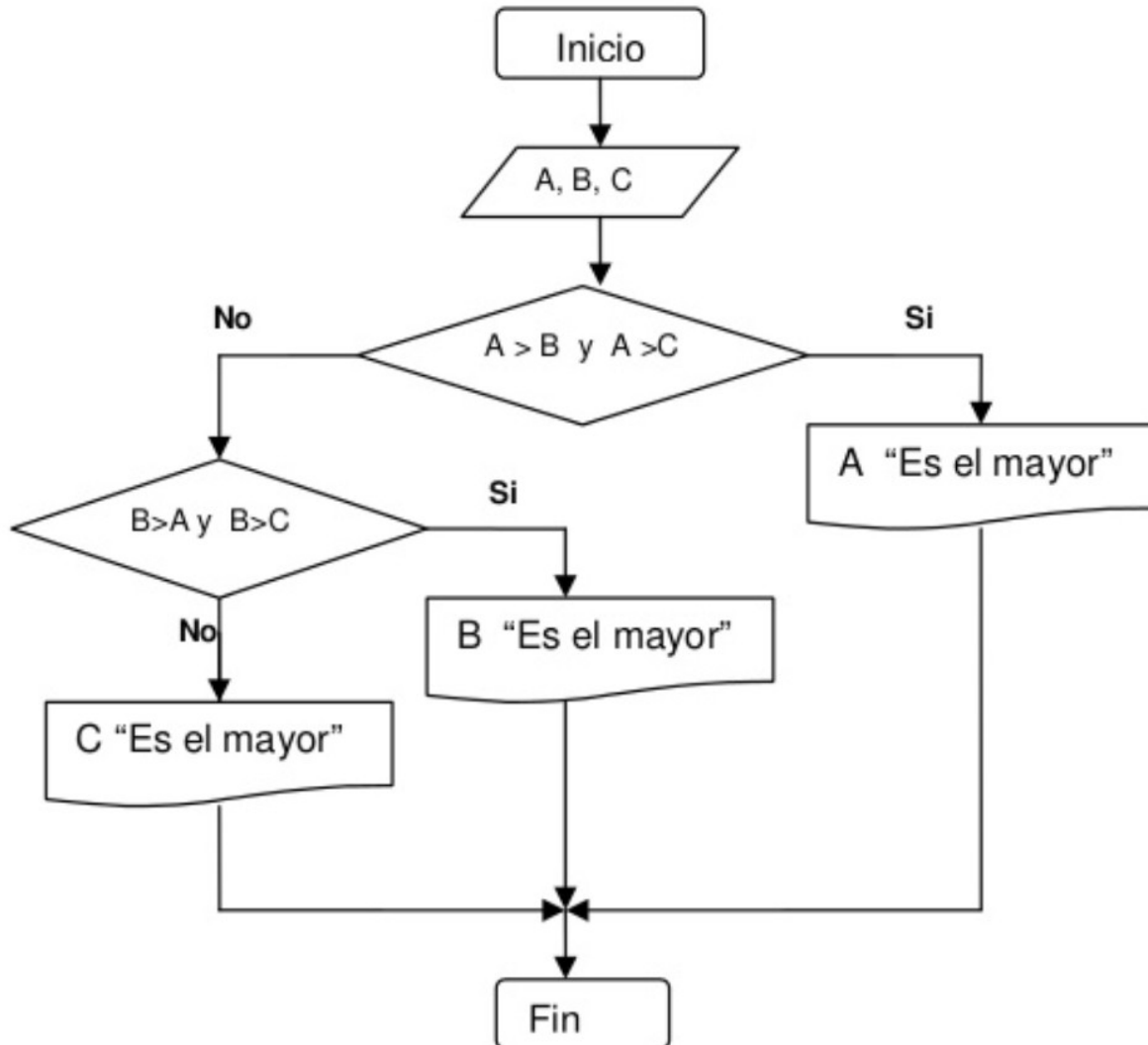
```
public class Principal {
    public static void main(String args[]) {
        int num = 20;
        int limite = num + 50;

        for(int i = num; i <= limite; i++){
            if(i != 12){
                if (i != 26){
                    if (i != 44){
                        if (i != 56){
                            System.out.println(i);
                        }
                    }
                }
            }
        }
    }
}
```



Ejercicio 6 (2 puntos)

Dado el siguiente diagrama de flujo, desarrolla el algoritmo en pseudocódigo que le corresponde:



Pseudocódigo

INICIO

Leer A

Leer B

Leer C

Si $A > B \ \&\& \ A > C$ **Hacer**
 Mostrar A + " Es el mayor"

Sino

Si $B > A \ \&\& \ B > C$ **Hacer**
 Mostrar B + " Es el mayor"

Sino

Mostrar C + " Es el mayor"

Fin Si

Fin Si

FIN