## PROGRAMACIÓN AVANZADA

**Profesores:** Cristian Chiang - Loreto Telgie Primer Semestre 2019

# **PROPÓSITO**

Al finalizar el curso el estudiante logrará describir conceptos, formular algoritmos y soluciones a problemas aplicando el paradigma de la programación orientada al objeto, utilizando estructuras de datos (arreglos y listas con nexo) y técnicas de resolución de problemas como recursión, dividir para conquistar y backtracking. El lenguaje de programación utilizado es Java.

#### RESULTADOS DE APRENDIZAJE

Los estudiantes al finalizar la asignatura deben ser capaces de:

- 1. Construir software orientado al objeto, legible y documentado, incluyendo testing (casos de prueba y prueba unitaria) y control de excepciones
- 2. Construir contenedores genéricos implementados con arreglos y con nexos
- 3. Resolver problemas utilizando contenedores ya implementados en Java (arraylist, linkedlist) e iteradores
- 4. Resolver problemas utilizando técnicas de recursión, dividir para conquistar y backtracking

### **CONTENIDO**

- 1. Introducción
- 2. Programación orientada al objeto en Java
- 3. Colecciones: Listas con nexo, Contenedores implementados en java (arraylist y linkedlist) e Iteradores
- 4. Estrategias de diseño de algoritmos: Recursión, Dividir para Conquistar y Backtracking

#### **APUNTES**

http://www.ucn.cl/onlineucn (En la plataforma EDUC@ v2.0)

#### **BIBLIOGRAFIA**

#### Textos Guía:

- Objects First with Java, David J. Barnes & Michael Kölling, Fifth edition, Prentice Hall / Pearson Education, 2012.
- Loreto Telgie. Apuntes de la asignatura Programación Avanzada, 2016

## **Texto Compelmentario**

• Data Structures and Problem Solving Using Java. Weiss, Mark Allen. Fourth edition, Addison-Wesley, 2009.

Máximo 5 minutos de atraso en la llegada a la clase

Requisito de asistencia a cátedra: 70%

Requisito de asistencia a ayudantías en laboratorio: 70% (5 de 7)

Requisito de asistencia a ayudantías en sala: 70% (5 de 7)

E	KIGEN	CIAS			2			
• 3 Pruebas de Cátedra								
		P0: P1: P2:	Sa 25 Mayo, 9:55 hrs       R         Sa 15 Junio, 9:55 hrs       R	Vivelación RA1 y parte del RA2 RA1, resto del RA2 y RA3	*			
		P3:	Ju 11 Julio, 9:55 hrs. R	2A4	Capítulo4			
		Examen	Recuperativo: Mi 17 Julio 9:55 hrs					
•	con no debe e	ota >= 4,0 estar en in	atorios de grupos de a 2 estudiantes d . Todas las entregas de los talleres se glés. Enunciado del Taller 3 en inglés e o Netbeans	on obligatorias. La doc				
		T1	Java con arreglos y subprogramas (sin	<u> </u>	i 20 Marzo – Ma 16 Abril			
			Análogo al último taller de introducció Entrega 1: Estructuras para los datos ( Entrega 2: Estructura del programa (1 Entrega 3: Código (70%)	(15%)	Ma 26 Marzo Lu 1 Abril Ma 16 Abril			
		T2:	Java sin herencia, navegabilidad y con	tenedores implementados	s con arreglos.			
			Se usa arquitectura propuesta	· · · · · · · · · · · · · · · · · · ·				
			Entrega 1 (15%): Modelo del dominio + Diagrama de clases dominio aplicación					
			F. (2004) G. F.	1 (7)	Ma 23 Abril			
			Entrega 2 (20%): Contratos + Diagram Entrega Final (65%)	na de Clases	Ma 30 Abril Vi 10 Mayo			
		T3:	lava con herencia, polimorfismo, navegabilidad y contene Con arquitectura		implementados con arreglos <b>0 Mayo - Ju 6 Junio</b>			
			Entrega 1 (10%): Modelo del dominio	+ contratos (10%) Vi 1	7 Mayo			
			Entrega 2 (15%): Diagrama de Clases		4 Mayo			
			Entrega Final (75%): Código	Mi 5	Junio			
		T4:	Java con herencia, polimorfismo, navegabilidad y contenedores implementados con lis nexo y uso de arraylist, linkedlist e iteradores. Con arquitectura		a			
			Entrega 1 (25%): Modelo del dominio Entrega Final (75%): Código	+ contratos + DC Mi 1	5 <b>Junio – Ma 25 Junio</b> 2 Junio 25 Junio			
		T5:	Recursión, Dividir para conquistar y B	Backtracking Ma	25 Junio – Lu 22 Julio			
			Entrega 1: Códigos recursivos, incluye	<u> </u>				
			Entropo 2. Madalandari 11 11		Julio			
			Entrega 2: Modelamiento del problema Entrega 3: Código del problema de bad Todas las entregas con la misma pondo	ctracking Lu 2	2 Julio 2 Julio			
			1 odas ias entregas con la misma ponde	Cracion				

Nota: Inmediatamente después de cada entrega, se publica una alternativa de solución

### **Promedio Talleres**

Si  $T2 \ge 4.0 \text{ y } T3 \ge 4.0 \text{ y } T4 \ge 4.0 \text{ y } T5 \ge 4.0$ 

Prom. Talleres = (T1 + T2 + T3 + T4 + T5) / 5

Sino

PromTalleres = mínimo{T1, T2, T3, T4, T5}

## No hay recuperativa por reprobación de talleres

Grupos de talleres: 2 estudiantes del mismo paralelo

### Ayudantes de Talleres: Michael Bravo-Brian Pardo

- Diseñar y corregir talleres, coordinado con Cristian Chiang
- Atender alumnos por consultas de talleres
- Atender alumnos por reclamos de talleres

Ayudante de corrección Loreto: Juan Carlos Lamas Ayudante de corrección Cristian: Juan Martinez

• Corregir trabajos clases y controles

## Ayudante de Cátedra (Laboratorio/Sala): Loreto: Juan Martinez Cristian:

- Diseño de ayudantías y laboratorios coordinado con Loreto
- Hacer ayudantías y laboratorios

Planificación Avudantías de Cátedra (Sala/Laboratorios): Viernes F (paralelo Loreto)

#	Fecha	Objetivo	Apoyo a	S/L
1	Vi15 Marzo	Ruteo de programa Java	Nivelación Nivelación	Laboratorio
		<ul> <li>Instalación y uso de Java, Eclipse o Netbeans</li> </ul>		
		Construir programas Java sencillos:		
		➤ Imprimir "hola mundo"		
		Debugging		
2	Vi 22	Construir programas Java con:	Nivelación	Laboratorio
	Marzo	Con arreglos		
		Deben partir dibujando la estructura de los datos		
		y la estructura del programa		
		• Hagan uso de algoritmos de ordenamiento,		
		búsqueda, inserción, etc.		
		Debugging		
3	Vi 29	Construir programas Java con:	Nivelación	Laboratorio
	Marzo	Con arreglos		
		Deben partir dibujando la estructura de los datos		
		y la estructura del programa		
		• Hagan uso de algoritmos de ordenamiento,		
		búsqueda, inserción, etc.		
		Hagan uso de archivos y subprogramas		
		Debugging		

Ejercicios pequeños:   Modelo del dominio   Diagrama de clases del dominio de la aplicación. con sus atributos, constructor y métodos get y set.   Sin herencia, ni asociaciones entre clases   Construcción del código de las clases y main creando objetos   Uso de contenedores   Sin arquitectura   Ejemplot: Listas de objetos		T	T	T_ :	
Diagrama de clases del dominio de la aplicación, con sus atributos, constructor y métodos get y set. Sin herencia, ni asociaciones entre clases Construcción del código de las clases y main creando objetos Uso de contenedores Sin arquitectura Ejemplo: Listas de objetos Ejercicio spequeños: Diagrama de clases Con contenedores Con contenedores Con contenedores Con arquitectura propuesta Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con contenedores Con contenedores Con contenedores Con raquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Ejercicio simple, incluyendo todo Dista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Prueba 2 Sala  Vi 7 Junio Vi 7 Junio Vi 14 Junio Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo. Prueba 2 Laboratorio Linkedlist, interface list, Iteradores  Vi 21 Junio Rue de algoritmos recursivos Algoritmos recursivos Algoritmos recursivos Sin dividir para conquistar Prueba 3 Laboratorio Modelamiento de problema de bactracking Prueba 3 Laboratorio	4	Vi 5 Abril		Prueba 1	Sala
con sus artibutos, constructor y métodos get y set.  Sin herencia, ni asociaciones entre clases  Construcción del código de las clases y main creando objetos  Uso de contenedores Sin arquitectura Ejemplo: Listas de objetos  Ejercicios pequeños: Diagrama de clases Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A Ejercicio tipo prueba, incluyendo todo Prueba 1 Laboratorio Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala Prueba 2 Sala  10 Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Prueba 2 Sala  10 Vi 7 Junio Prueba 2 Laboratorio Linkedlist, interface list, Iteradores Vi 21 Junio Rubolema simples que utilicen listas con nexo. Prueba 3 Laboratorio Algoritmos recursivos Algoritmos recursivos Sin dividir para conquistar Prueba 3 Laboratorio Dividir para conquistar Prueba 3 Laboratorio					
Sin herencia, ni asociaciones entre clases Construcción del código de las clases y main creando objetos Uso de contenedores Sin arquitectura Ejemplo: Listas de objetos  Sin berencia Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio grande: Modelo del dominio Diagrama de clases Sin herencia Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con contenedores Con arquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A Vi 17 Mayo Ejercicio ipo prueba, incluyendo todo Prueba 1 Sala  Vi 24 Mayo Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Vi 21 Junio Vi 7 Junio Problema cino listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Vi 21 Junio Vi 24 Junio Vi 25 Junio Ruteo de algoritmos recursivo Algoritmos recursivo Sin dividir para conquistar Prueba 3 Laboratorio Nodelamiento de problema de bactracking  Vi 28 Junio Construcción del código del problema de Prueba 3 Laboratorio					
Construcción del código de las clases y main creando objetos     Uso de contenedores     Sin arquitectura Ejemplo: Listas de objetos     Diagrama de clases     Con asociaciones entre clases     Sin herencia     Con contenedores     Con arquitectura propuesta     Diagrama de clases     Con arquitectura propuesta     Sala      Vi 3 Mayo     Modelo del dominio     Diagrama de clases del dominio de la aplicación     Con asociaciones entre clases     Sin herencia     Con contenedores     Con arquitectura propuesta     Sala      Vi 10 Mayo     Vi 10 Mayo     Vi 10 Mayo     Vi 10 Mayo     Vi 24 Mayo     De Con arquitectura propuesta     De Con arquitectura     De Con arquitectura     De Prueba 1 Laboratorio      Vi 24 Mayo     Problemas simples que utilicen listas con un nexo.     Incluir:     Diagrama de clases     Código para la clase nodo y clase lista.     Uso de arquitectura     Diagrama de clases     Código para la clase nodo y clase lista.     Uso de arquitectura     Prueba 2 Sala  10 Vi 7 Junio     Prueba 2 Laboratorio     Nalorimors recursivo     Algorimors recursivo     Algorimors recursivos     Algorimors recursivos     Sin dividir para conquistar     Oritri para conquistar     Prueba 3 Laboratorio     Nodelamiento de problema de bactracking					
creando objetos  • Uso de contenedores • Sin arquitectura Ejemplo: Listas de objetos  Bjercicios pequeños: • Con asociaciones entre clases • Sin herencia • Con contenedores • Con arquitectura propuesta  6 Vi 3 Mayo  Bjercicio grande: • Modelo del dominio • Diagrama de clases del dominio de la aplicación • Con asociaciones entre clases • Sin herencia • Con contenedores • Con arquitectura propuesta • Con contenedores • Con arquitectura propuesta • Con contenedores • Con contenedores • Con arquitectura propuesta • Con contenedores • Con arquitectura propuesta • Con contenedores • Con arquitectura propuesta • Vi 10 Mayo  Fjercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  8 Vi 17 Mayo  Forblemas simples que utilicen listas con un nexo. Incluir: • Diagrama de clases • Código para la clase nodo y clase lista. • Uso de arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  10 Vi 7 Junio Problema con listas con nexo.  11 Vi 14 Junio Problema ton listas con nexo. • Ruteo de algoritmos recursivos • Algoritmos recursivos • Algoritmos recursivos • Sin dividir para conquistar • Dividir para conquistar • Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio			· ·		
Viso de contenedores     Sin arquitectura     Ejemplo: Listas de objetos     Diagrama de clases     Con asociaciones entre clases     Sin herencia     Con contenedores     Con arquitectura propuesta     Ejercicio parante:     Modelo del dominio     Diagrama de clases     Modelo del dominio     Diagrama de clases del dominio de la aplicación     Con asociaciones entre clases     Sin herencia     Con contenedores     Con arquitectura propuesta     Ejercicio grande:     Modelo del dominio     Diagrama de clases del dominio de la aplicación     Con contenedores     Con contenedores     Con contenedores     Con arquitectura propuesta     Con contenedores     Con arquitectura propuesta     Ejercicio simple, incluyendo todo     Con arquitectura propuesta de la prueba la clase de lobjetos, donde el objetos puede ser de tipo Al o A2, ambos subclases de A      Vi 17 Mayo     Ejercicio tipo prueba, incluyendo todo     Vi 24 Mayo     Vi 24 Mayo     Problemas simples que utilicen listas con un nexo. Incluir:     Diagrama de clases     Código para la clase nodo y clase lista.     Uso de arquitectura     Diagrama de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura     El mismo problema fe la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura     El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura     El mismo problema fe la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura     El mismo problema fe la grandina de la condition en listas con nexo, usando cualquier variación en listas con nexo, usando cualquier vari					
Sin arquitectura Ejemplo: Listas de objetos Ejercicios pequeños: Diagrama de clases Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A Sil Vi 17 Mayo Fjercicio tipo prueba, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Codigo para la clase nodo y clase lista. Uso de arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo, usando arquitectura El mismo problema de la contro secursivo Algoritmos recursivos Algoritmos recursivos Algoritmos recursivos Algoritmos recursivos Algoritmos recursivos Algoritmos recursivos Sin dividir para conquistar Modelamiento de problema de bactracking  14 Vi 25 Julio Construcción del código del problema de Prueba 3 Laboratorio					
Fjemplo: Listas de objetos   Prueba 1   Sala			Uso de contenedores		
Sala			=		
Diagrama de clases Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta  Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con asociaciones entre clases Sin herencia Con contenedores Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Fiercicio tipo prueba, incluyendo todo Prueba 1 Sala  Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores Algoritmos recursivo Algoritmos recursivo Algoritmos recursivo Sin dividir para conquistar Modelamiento de problema de bactracking  Prueba 3 Laboratorio					
Con asociaciones entre clases Sin herencia Con contenedores Con arquitectura propuesta  Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con contenedores Con arquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala  Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Vi 21 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores Algoritmos recursivo Sin dividir para conquistar Nodelamiento de problema de Prueba 3 Laboratorio Construcción del código del problema de Prueba 3 Laboratorio	5	Vi 26 Abril		Prueba 1	Sala
Sin herencia Con contenedores Con arquitectura propuesta  Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con contenedores Con arquitectura propuesta Con contenedores Con arquitectura propuesta  7 Vi 10 Mayo Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo A1 o A2, ambos subclases de A  8 Vi 17 Mayo Problema simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  10 Vi 7 Junio Problema tel la yudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio Problema tel la yudantía anterior, usando cualquier variación en listas con nexo. Prueba 2 Laboratorio Linkedlist, interface list, Iteradores  12 Vi 21 Junio Ruteo de algoritmos recursivos Algoritmos recursivos Algoritmos recursivo Sin dividir para conquistar Nodelamiento de problema de bactracking  Prueba 3 Laboratorio					
Con contenedores Con arquitectura propuesta  Ejercicio grande: Modelo del dominio Diagrama de clases del dominio de la aplicación Con asociaciones entre clases Sin herencia Con contenedores Con tratos DC Con arquitectura propuesta Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala  Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  In Vi 14 Junio Problema con listas con nexo. Linkedlist, interface list, Iteradores  Prueba 2 Laboratorio  Laboratorio  Vi 21 Junio Ruteo de algoritmos recursivos Algoritmos recursivos Sin dividir para conquistar Modelamiento de problema de Drueba 3 Laboratorio  Laboratorio  Construcción del código del problema de Prueba 3 Laboratorio			<ul> <li>Con asociaciones entre clases</li> </ul>		
• Con arquitectura propuesta  Ejercicio grande: • Modelo del dominio • Diagrama de clases del dominio de la aplicación • Con asociaciones entre clases • Sin herencia • Con contenedores • Contratos • DC • Con arquitectura propuesta  Frueba 1  Vi 10 Mayo  Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo  Ejercicio tipo prueba, incluyendo todo Prueba 1  Ejercicio tipo prueba, incluyendo todo Prueba 1  Sala  Prueba 2  Sala  Vi 24 Mayo  Problemas simples que utilicen listas con un nexo. Incluir: • Diagrama de clases • Código para la clase nodo y clase lista. • Uso de arquitectura  In Vi 7 Junio  Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  In Vi 14 Junio  Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  Prueba 2  Laboratorio  Laboratorio  Algoritmos recursivo • Sin dividir para conquistar • Modelamiento de problema de bactracking  Prueba 3  Laboratorio			Sin herencia		
Vi 3 Mayo   Ejercicio grande:   Modelo del dominio   Diagrama de clases del dominio de la aplicación   Con asociaciones entre clases   Sin herencia   Con contenedores   Con tratos   DC   Con arquitectura propuesta     7			Con contenedores		
Modelo del dominio     Diagrama de clases del dominio de la aplicación     Con asociaciones entre clases     Sin herencia     Con contenedores     Contratos     DC     Con arquitectura propuesta     Ejercicio simple, incluyendo todo     Lista de objetos, donde el objeto puede ser de tipo     Al o A2, ambos subclases de A      Vi 17 Mayo     Ejercicio tipo prueba, incluyendo todo     Prueba 1     Sala      Vi 24 Mayo     Problemas simples que utilicen listas con un nexo.     Incluir:     Diagrama de clases     Código para la clase nodo y clase lista.     Uso de arquitectura     Problema con listas con nexo, usando arquitectura     El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio     Problema tipo prueba, incluyendo Arraylist,     Linkedlist, interface list, Iteradores     Algoritmos recursivos     Algoritmos recursivos     Algoritmos recursivos     Sin dividir para conquistar     Modelamiento de problema de bactracking  14 Vi 5 Julio     Construcción del código del problema de Prueba 3     Laboratorio			Con arquitectura propuesta		
Diagrama de clases del dominio de la aplicación     Con asociaciones entre clases     Sin herencia     Con contenedores     Contratos     DC     Con arquitectura propuesta      Prueba 1     Laboratorio      B Vi 10 Mayo     Ejercicio simple, incluyendo todo     Lista de objetos, donde el objeto puede ser de tipo     Al o A2, ambos subclases de A     Ejercicio tipo prueba, incluyendo todo     Prueba 1     Sala     Problemas simples que utilicen listas con un nexo.     Incluir:	6	Vi 3 Mayo	Ejercicio grande:	Prueba 1	Sala
Con asociaciones entre clases Sin herencia Con contenedores Contratos DC Con arquitectura propuesta  Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Vi 21 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores Algoritmos recursivos Algoritmos recursivos Sin dividir para conquistar Modelamiento de problema de bactracking  Prueba 3 Laboratorio  Laboratorio  Prueba 3 Laboratorio  Prueba 3 Laboratorio  Alaboratorio  Laboratorio  Prueba 3 Laboratorio			Modelo del dominio		
Con asociaciones entre clases Sin herencia Con contenedores Contratos DC Con arquitectura propuesta  Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala Problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Vi 21 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores Algoritmos recursivos Algoritmos recursivos Sin dividir para conquistar Modelamiento de problema de bactracking  Prueba 3 Laboratorio  Laboratorio  Prueba 3 Laboratorio  Prueba 3 Laboratorio  Alaboratorio  Laboratorio  Prueba 3 Laboratorio			<ul> <li>Diagrama de clases del dominio de la aplicación</li> </ul>		
Con contenedores     Contratos     DC     Con arquitectura propuesta      Vi 10 Mayo     Ejercicio simple, incluyendo todo     Lista de objetos, donde el objeto puede ser de tipo     Al o A2, ambos subclases de A      Vi 24 Mayo     Prueba 1     Sala     Prueba 2     Sala      Vi 24 Mayo     Problemas simples que utilicen listas con un nexo.     Incluir:          Diagrama de clases          Código para la clase nodo y clase lista.          Uso de arquitectura          Uso de arquitectura     Problema con listas con nexo, usando arquitectura     El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio     Problema tipo prueba, incluyendo Arraylist,     Linkedlist, interface list, Iteradores  12 Vi 21 Junio     Ruteo de algoritmos recursivos     Algoritmos recursivo     Sin dividir para conquistar     Nodelamiento de problema de bactracking  14 Vi 5 Julio     Construcción del código del problema de Prueba 3     Laboratorio			1		
Con contenedores     Contratos     DC     Con arquitectura propuesta      Vi 10 Mayo     Ejercicio simple, incluyendo todo     Lista de objetos, donde el objeto puede ser de tipo     Al o A2, ambos subclases de A      Vi 24 Mayo     Prueba 1     Sala     Prueba 2     Sala      Vi 24 Mayo     Problemas simples que utilicen listas con un nexo.     Incluir:          Diagrama de clases          Código para la clase nodo y clase lista.          Uso de arquitectura          Uso de arquitectura     Problema con listas con nexo, usando arquitectura     El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio     Problema tipo prueba, incluyendo Arraylist,     Linkedlist, interface list, Iteradores  12 Vi 21 Junio     Ruteo de algoritmos recursivos     Algoritmos recursivo     Sin dividir para conquistar     Nodelamiento de problema de bactracking  14 Vi 5 Julio     Construcción del código del problema de Prueba 3     Laboratorio					
Contratos DC Con arquitectura propuesta  Figural propuesta  Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo Al o A2, ambos subclases de A  Vi 17 Mayo Figural problemas simples que utilicen listas con un nexo. Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura  Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Prueba 2  Sala  Prueba 2  Sala  Vi 7 Junio Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Prueba 2  Laboratorio  Linkedlist, interface list, Iteradores  Linkedlist, interface list, Iteradores Algoritmos recursivo Algoritmos recursivo Sin dividir para conquistar Nodelamiento de problema de bactracking  Prueba 3  Laboratorio					
<ul> <li>DC</li></ul>					
<ul> <li>Con arquitectura propuesta</li> <li>Vi 10 Mayo</li> <li>Ejercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo A1 o A2, ambos subclases de A</li> <li>Vi 17 Mayo</li> <li>Ejercicio tipo prueba, incluyendo todo</li> <li>Prueba 1</li> <li>Sala</li> <li>Vi 24 Mayo</li> <li>Problemas simples que utilicen listas con un nexo. Incluir: <ul> <li>Diagrama de clases</li> <li>Código para la clase nodo y clase lista.</li> <li>Uso de arquitectura</li> </ul> </li> <li>Prueba 2</li> <li>Sala</li> </ul> <li>Vi 7 Junio</li> <li>Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.</li> <li>Vi 14 Junio</li> <li>Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores</li> <li>Prueba 2</li> <li>Laboratorio</li> <li>Algoritmos recursivos  <ul> <li>Algoritmos recursivos</li> <li>Sin dividir para conquistar</li> <li>Prueba 3</li> </ul> </li> <li>Vi 28 Junio</li> <li>Dividir para conquistar  <ul> <li>Dividir para conquistar</li> <li>Modelamiento de problema de bactracking</li> </ul> </li> <li>Prueba 3</li> <li>Laboratorio</li> <li>Laboratorio</li>					
7Vi 10 MayoEjercicio simple, incluyendo todo Lista de objetos, donde el objeto puede ser de tipo A1 o A2, ambos subclases de APrueba 1Laboratorio8Vi 17 MayoEjercicio tipo prueba, incluyendo todoPrueba 1Sala9Vi 24 MayoProblemas simples que utilicen listas con un nexo. Incluir: <ul><li>Diagrama de clases </li></ul>					
Lista de objetos, donde el objeto puede ser de tipo A1 o A2, ambos subclases de A  8 Vi 17 Mayo Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala  9 Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir:  • Diagrama de clases • Código para la clase nodo y clase lista. • Uso de arquitectura  10 Vi 7 Junio Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  12 Vi 21 Junio • Ruteo de algoritmos recursivos • Algoritmos recursivo • Algoritmos recursivo • Sin dividir para conquistar • Dividir para conquistar • Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio	7	Vi 10 Mayo	* * *		Laboratorio
A1 o A2, ambos subclases de A  8 Vi 17 Mayo Ejercicio tipo prueba, incluyendo todo Prueba 1 Sala  9 Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir:  • Diagrama de clases  • Código para la clase nodo y clase lista.  • Uso de arquitectura  10 Vi 7 Junio Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  12 Vi 21 Junio • Ruteo de algoritmos recursivos  • Algoritmos recursivo  • Sin dividir para conquistar  • Dividir para conquistar  • Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio		, 1 10 1.1 <b>0</b> 0			200 0100110
8Vi 17 MayoEjercicio tipo prueba, incluyendo todoPrueba 1Sala9Vi 24 MayoProblemas simples que utilicen listas con un nexo. Incluir:			, , , , , , , , , , , , , , , , , , ,		
9 Vi 24 Mayo Problemas simples que utilicen listas con un nexo. Incluir:  • Diagrama de clases • Código para la clase nodo y clase lista. • Uso de arquitectura  Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  12 Vi 21 Junio • Ruteo de algoritmos recursivos • Algoritmos recursivo • Sin dividir para conquistar  13 Vi 28 Junio • Dividir para conquistar • Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio	8	Vi 17 Mayo		Prueba 1	Sala
Incluir: Diagrama de clases Código para la clase nodo y clase lista. Uso de arquitectura Problema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  Prueba 2 Sala  Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  Vi 21 Junio Ruteo de algoritmos recursivos Algoritmos recursivo Sin dividir para conquistar  Vi 28 Junio Dividir para conquistar  Prueba 3 Laboratorio  Sala  Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio		•		Prueba 2	Sala
<ul> <li>Código para la clase nodo y clase lista.</li> <li>Uso de arquitectura</li> <li>Problema con listas con nexo, usando arquitectura         El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.</li> <li>Vi 14 Junio</li> <li>Problema tipo prueba, incluyendo Arraylist,         Linkedlist, interface list, Iteradores</li> <li>Vi 21 Junio</li> <li>Ruteo de algoritmos recursivos         <ul> <li>Algoritmos recursivo</li> <li>Sin dividir para conquistar</li> </ul> </li> <li>Vi 28 Junio</li> <li>Dividir para conquistar</li> <li>Modelamiento de problema de bactracking</li> <li>Vi 5 Julio</li> <li>Construcción del código del problema de Prueba 3</li> <li>Laboratorio</li> </ul>			· · · · · · · · · · · · · · · · · · ·		
<ul> <li>Código para la clase nodo y clase lista.</li> <li>Uso de arquitectura</li> <li>Problema con listas con nexo, usando arquitectura         El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.</li> <li>Vi 14 Junio</li> <li>Problema tipo prueba, incluyendo Arraylist,         Linkedlist, interface list, Iteradores</li> <li>Vi 21 Junio</li> <li>Ruteo de algoritmos recursivos         <ul> <li>Algoritmos recursivo</li> <li>Sin dividir para conquistar</li> </ul> </li> <li>Vi 28 Junio</li> <li>Dividir para conquistar</li> <li>Modelamiento de problema de bactracking</li> <li>Vi 5 Julio</li> <li>Construcción del código del problema de Prueba 3</li> <li>Laboratorio</li> </ul>			Diagrama de clases		
<ul> <li>Uso de arquitectura</li> <li>Problema con listas con nexo, usando arquitectura</li> <li>El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.</li> <li>Vi 14 Junio</li> <li>Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores</li> <li>Vi 21 Junio</li> <li>Ruteo de algoritmos recursivos</li> <li>Algoritmos recursivo</li> <li>Sin dividir para conquistar</li> <li>Vi 28 Junio</li> <li>Dividir para conquistar</li> <li>Modelamiento de problema de bactracking</li> <li>Vi 5 Julio</li> <li>Construcción del código del problema de Prueba 3</li> <li>Laboratorio</li> </ul>			8		
10Vi 7 JunioProblema con listas con nexo, usando arquitectura El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.Prueba 2Sala11Vi 14 JunioProblema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, IteradoresPrueba 2Laboratorio12Vi 21 Junio• Ruteo de algoritmos recursivos • Algoritmos recursivo • Sin dividir para conquistarPrueba 3Laboratorio13Vi 28 Junio• Dividir para conquistar • Modelamiento de problema de bactrackingPrueba 3Sala14Vi 5 JulioConstrucción del código del problema de Prueba 3Laboratorio					
El mismo problema de la ayudantía anterior, usando cualquier variación en listas con nexo.  11 Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  12 Vi 21 Junio • Ruteo de algoritmos recursivos • Algoritmos recursivo • Sin dividir para conquistar  13 Vi 28 Junio • Dividir para conquistar • Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio	10	Vi 7 Junio	*	Prueba 2	Sala
cualquier variación en listas con nexo.  11 Vi 14 Junio Problema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, Iteradores  12 Vi 21 Junio • Ruteo de algoritmos recursivos • Algoritmos recursivo • Sin dividir para conquistar  13 Vi 28 Junio • Dividir para conquistar • Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio			, · · · · · · · · · · · · · · · · · · ·		
11Vi 14 JunioProblema tipo prueba, incluyendo Arraylist, Linkedlist, interface list, IteradoresPrueba 2Laboratorio12Vi 21 Junio• Ruteo de algoritmos recursivos • Algoritmos recursivo • Sin dividir para conquistarPrueba 3Laboratorio13Vi 28 Junio• Dividir para conquistar • Modelamiento de problema de bactrackingPrueba 3Sala14Vi 5 JulioConstrucción del código del problema de Prueba 3Laboratorio			· · · · · · · · · · · · · · · · · · ·		
Linkedlist, interface list, Iteradores  12 Vi 21 Junio  Ruteo de algoritmos recursivos  Algoritmos recursivo  Sin dividir para conquistar  Vi 28 Junio  Dividir para conquistar  Modelamiento de problema de bactracking  14 Vi 5 Julio  Laboratorio  Prueba 3  Sala  Prueba 3  Laboratorio	11	Vi 14 Junio		Prueba 2	Laboratorio
12Vi 21 Junio• Ruteo de algoritmos recursivos • Algoritmos recursivo • Sin dividir para conquistarPrueba 3Laboratorio13Vi 28 Junio • Modelamiento de problema de bactrackingPrueba 3Sala14Vi 5 JulioConstrucción del código del problema de Prueba 3Laboratorio			<u> </u>		
<ul> <li>Algoritmos recursivo</li> <li>Sin dividir para conquistar</li> <li>Vi 28 Junio</li> <li>Dividir para conquistar</li> <li>Modelamiento de problema de bactracking</li> <li>Vi 5 Julio</li> <li>Construcción del código del problema de Prueba 3</li> <li>Laboratorio</li> </ul>	12	Vi 21 Junio		Prueba 3	Laboratorio
<ul> <li>Sin dividir para conquistar</li> <li>Vi 28 Junio</li> <li>Dividir para conquistar</li> <li>Modelamiento de problema de bactracking</li> <li>Vi 5 Julio</li> <li>Construcción del código del problema de Prueba 3</li> <li>Laboratorio</li> </ul>					
13Vi 28 Junio• Dividir para conquistar • Modelamiento de problema de bactrackingPrueba 3Sala14Vi 5 JulioConstrucción del código del problema de Prueba 3Laboratorio					
Modelamiento de problema de bactracking  14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio	13	Vi 28 Junio		Prueba 3	Sala
14 Vi 5 Julio Construcción del código del problema de Prueba 3 Laboratorio		, 1 20 0 01110		1100000	~
	1.0	77' 6 T 1'	2	D 1 2	т 1
backtracking modelado la ayudantia anterior	14	V1 5 Julio		Prueba 3	Laboratorio
			backtracking moderado la ayudantia anterior		