



# Introducción a la Programación y Computación 2

*Ing. Francisco Yuman*  
*Diciembre 2023*



## Descripción del Curso

*Este curso está diseñado para que el estudiante inicie el proceso de modelado de sistemas de software utilizando los conceptos de la programación orientada a objetos, los ambientes en que estas soluciones son dispuestas, temas orientados a las bases de desarrollo sobre la web, Cloud, así como los temas asociados al manejo de versiones y control del software.*



# Unidades a Cubrir

---

- ✓ Estructura y manejo de la memoria
- ✓ Principios de Diseño de Software
- ✓ Arquitecturas para soluciones de Software
- ✓ Estrategias de Desarrollo de Software
- ✓ Security & Quality Assurance II

# Evaluación

---

- *Primer Parcial* : **13 puntos**
- *Segundo Parcial* : **13 puntos**
- *Tercer Parcial* : **13 puntos**
- *Tareas, otros* : **06 puntos**
- *Laboratorio* : **30 puntos**
- 
- *Zona* : **75 puntos**
- *Examen Final* : **25 puntos**
- 
- *Nota Final* : **100 puntos**

# Fechas Importantes

---

## Primer Parcial

*11 Diciembre 2023*

## Segundo Parcial

*18 Diciembre 2023*

## Tercer Parcial

*26 Diciembre 2023*

## Proyecto

*28 Diciembre 2023*

## Examen Final

*29 Diciembre 2023*

# Unidad I

Unidad	Tema
1. Estructura y manejo de la memoria	<ul style="list-style-type: none"><li>1. Modelos de administración de memoria<ul style="list-style-type: none"><li>1.1. Administración física de la memoria</li><li>1.2. Administración lógica de la memoria<ul style="list-style-type: none"><li>1.2.1. Reclamo de memoria</li><li>1.2.2. Dellocation Controlada</li><li>1.2.3. Manejo automático de la memoria</li><li>1.2.4. Garbage Collector</li></ul></li></ul></li><li>2. Manejo de Memoria dinámica<ul style="list-style-type: none"><li>2.1. Apuntadores<ul style="list-style-type: none"><li>2.1.1. Los índices y el apuntador simple</li><li>2.1.2. El apuntador subíndice</li><li>2.1.3. Almacenamiento</li></ul></li><li>2.2. Tipos de datos abstractos (TDA's)<ul style="list-style-type: none"><li>2.2.1. Concepto</li><li>2.2.2. Listas simples</li><li>2.2.3. Listas doblemente enlazadas</li><li>2.2.4. Listas circulares</li><li>2.2.5. Pilas</li><li>2.2.6. Colas</li><li>2.2.7. Listas ortogonales</li><li>2.2.8. Listas n-encadenadas</li><li>2.2.9. Ordenamientos</li></ul></li></ul></li></ul>



# Modelos de administración de memoria

The background of the slide features a blurred image of several business professionals in an office setting. In the foreground, a person in a white shirt is holding a pen over a document. To the right, another person in a dark suit is visible, holding a laptop and a coffee cup. The overall tone is professional and collaborative.

# Administración de Memoria

---

Es una tarea realizada por el sistema operativo que consiste en gestionar la jerarquía de memoria, en cargar y descargar procesos en memoria principal para que sean ejecutados. Para ello el sistema operativo gestiona lo que se conoce como MMU o Unidad de Administración de Memoria, el cual es un dispositivo hardware que transforma las direcciones lógicas en físicas.

Su trabajo es seguir la pista de que partes de la memoria estén en uso y cuales no lo están, con el fin de poder asignar memoria a los procesos cuando la necesiten, y recuperar esa memoria cuando dejen de necesitarla, así como gestionar el intercambio entre memoria principal y el disco cuando la memoria principal resulte demasiado pequeña para contener a todos los procesos



# Objetivos de la Gestión de Memoria

---

- Ofrecer a cada proceso un espacio lógico propio.
- Proporcionar protección entre los procesos.
- Permitir que los procesos compartan memoria.
- Maximizar el rendimiento del sistema

# Requisitos de la Gestión de Memoria

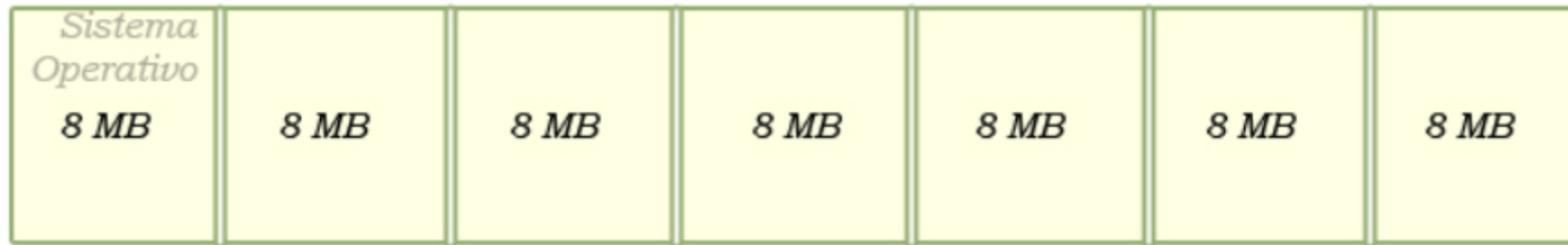
---

- **Reubicación:** En un sistema multiprogramado la memoria se encuentra compartida por varios procesos, por lo tanto, los procesos deben ser cargados y descargados de memoria.
- **Protección:** En un sistema con multiprogramación es necesario proteger al sistema operativo y a los otros procesos de posibles accesos que se puedan realizar a sus espacios de direcciones.
- **Organización Lógica:** Tanto la memoria principal como la secundaria presentan una organización física similar, como un espacio de direcciones lineal y unidimensional. Debe existir una cierta correspondencia entre el sistema operativo y el hardware al tratar los datos y los programas de los usuarios de acuerdo a la estructura lógica que ellos presenten.
- **Organización Física:** Debe ser parte de la administración de memoria, la organización del flujo de información entre la memoria principal y la memoria secundaria.

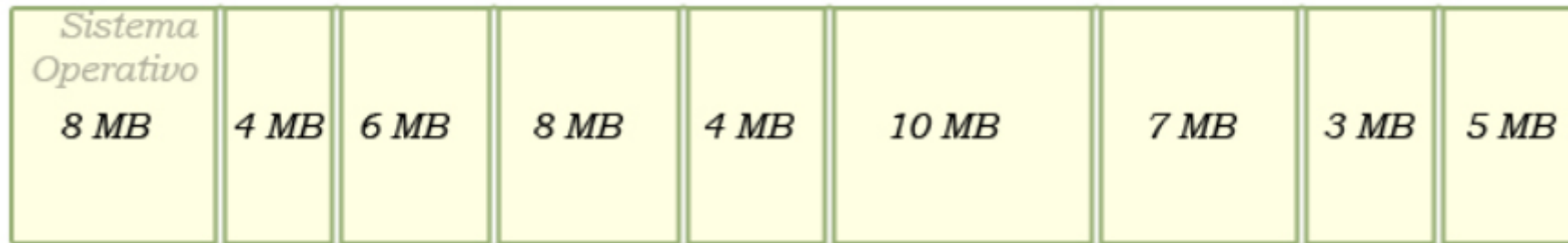
# Técnicas de Administración de Memoria (Asignación Contigua)

---

## Ejemplo de particiones estáticas



*Particiones de Igual Tamaño*



*Particiones de Distinto Tamaño*

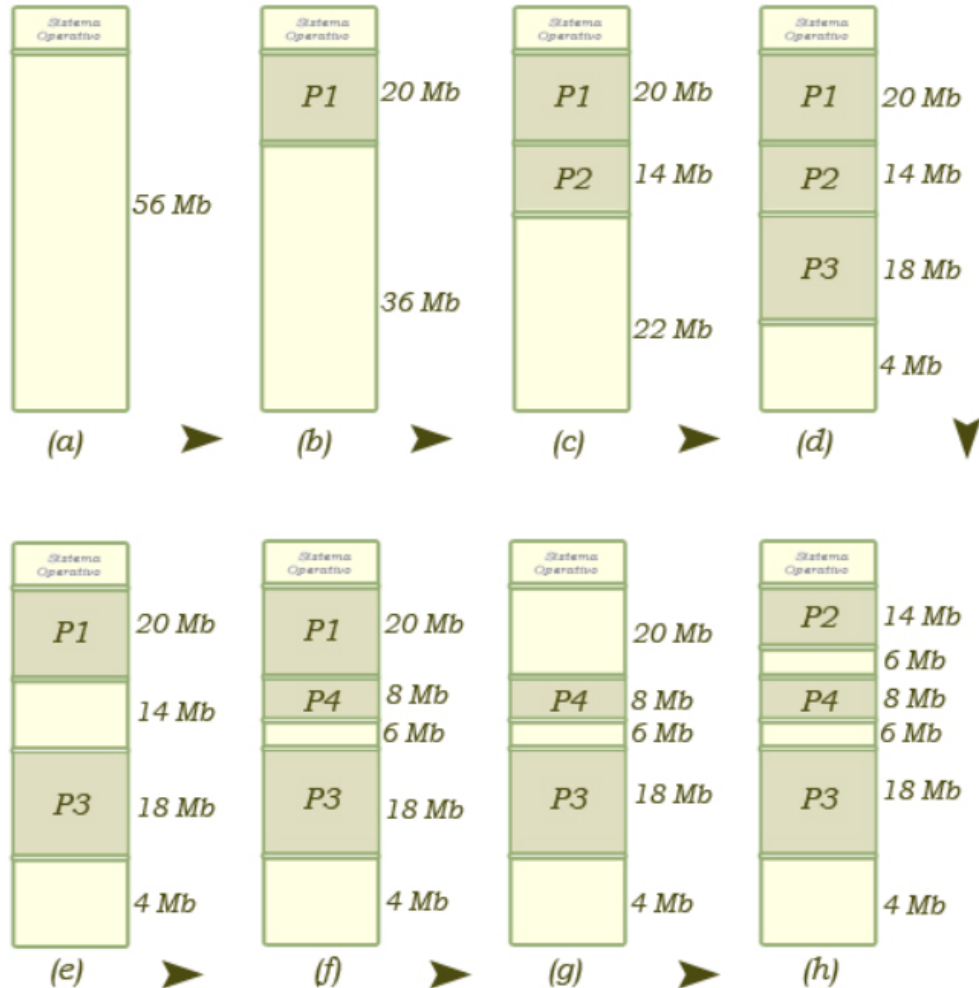
# Con particiones estáticas surgen dos dificultades:

---

- Un programa puede ser demasiado grande para caber en una partición, por lo tanto si el programa no se ha diseñado mediante superposición, simplemente no se puede ejecutar. De otro modo, podrán estar en memoria aquellos módulos del programa que se necesiten, pero se requerirá que estos módulos sean intercambiados a medida que la ejecución progresa.
- Se malgasta el espacio interno a cada partición cuando el bloque cargado es más pequeño, lo que se conoce como fragmentación interna. Es decir, cualquier proceso por pequeño que sea, ocupará una partición completa.



# Ejemplo de particiones dinámicas



En este ejemplo, partimos de la memoria libre completamente en (a), luego se cargan "P1" [20 MB], (b), "P2" [14 MB] (c) y "P3" [18 MB] (d). Se libera "P2" (e) y se carga "P4" [8 MB] (f), se libera "P1" (g) y se carga nuevamente "P2" [14 MB] (h). Notemos como se van reestructurando las particiones en base al tamaño de los procesos que se van cargando, esto sucede por ser particiones dinámicas.

# Algoritmos

---

- **Mejor ajuste:** consiste en ubicar el proceso en el espacio de memoria que mas se ajuste a su tamaño.
- **Primer ajuste:** consiste en ubicar el proceso en el primer hueco disponible, recorriendo desde el inicio de la memoria, cuyo tamaño sea suficiente para el proceso.
- **Peor ajuste:** consiste en ubicar el siguiente hueco disponible, que sea suficientemente grande, a partir de la última asignación de memoria.

# Ejercicio

---

Considerando que tiene una memoria que consiste en lo siguiente tamaños de espacios, por orden de memoria: 10 KB, 4 KB, 20 KB, 18 KB, 7 KB, 9 KB, 12 KB 15 KB.

¿Cuál espacio se toma para las siguientes solicitudes de segmento sucesivas:

- a) 12 KB
- b) 10 KB
- c) 9 KB

Simulando los algoritmos de primer ajuste, mejor ajuste y peor ajuste.

# Con particiones dinámicas surgen las siguientes dificultades:

---

- Producto de la entrada y salida de procesos en la memoria, se van generando porciones cada vez más pequeñas de la memoria sin utilizar, lo que se conoce como fragmentación externa.
- Para solucionar este problema se debe recurrir a la compactación de la memoria de manera de eliminar los espacios (huecos) entre procesos. Esto significa que los procesos deben ser reubicados en memoria en forma dinámica.



# Técnicas de Administración de Memoria (Asignación no Contigua)

---

- Dentro de la asignación no contigua, la administración de memoria implementa técnicas como Paginación y Segmentación

- Paginación

Esta técnica consiste en dividir la memoria en espacios de igual tamaño llamados páginas, en la memoria lógica y marcos de página en la memoria física. Mostramos una imagen de cómo se relacionan la memoria lógica con la memoria física a través de la Tabla de Páginas.

# Paginación

Dirección	MEMORIA LÓGICA	
0 - 4 k	A B C D	0
4 - 8 k	E F G H	1
8 - 12 k	I J K L	2
12-16 k	M N O P	3
16-20 k	Q R S T	4
20-24 k	U V W X	5
24-28 k	Y Z A1 B2	6
28-32 k	C3 D4 E5 F6	7
32-36 k	G7 H8 I9 J10	8
36-40 k	M11 N12 O13 P14	9
40-44 k	Q15 R16 S17 T18	10
44-48 k	U19 V20 W21 X22	11
48-52 k	Y23 Z24 A111 B222	12
52-56 k	C333 D444 E555 F666	13
56-60 k	G777 H888 I999 J 1000	14
60-64 k	K 2000 L 3000 M 4000 N 5000	15

Páginas

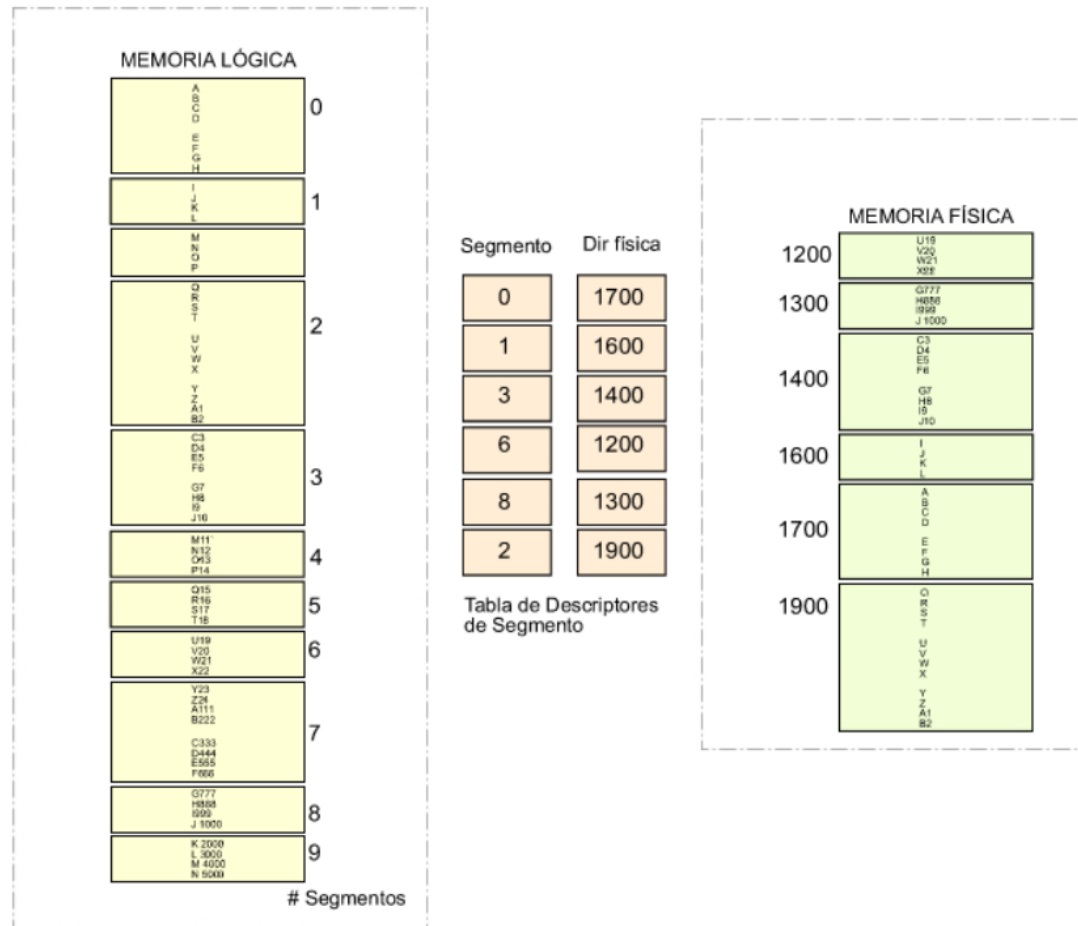
MARCO DE PÁGINAS	
0	5
1	6
2	3
3	X
4	7
5	X
6	X
7	2
8	4
9	X
10	X
11	0
12	X
13	X
14	1
15	X

Tabla de Página

Marco de Página	MEMORIA FÍSICA	
0 - 4 k	U19 V20 W21 X22	0
4 - 8 k	G777 H888 I999 J 1000	1
8 - 12 k	C3 D4 E5 F6	2
12-16 k	I J K L	3
16-20 k	G7 H8 I9 J10	4
20-24 k	A B C D	5
24-28 k	E F G H	6
28-32 k	O P Q R S T	7

# Segmentación

Consiste en dividir la memoria en espacios al igual que la Paginación, pero a diferencia de esta, la divide en espacios de diferente tamaño. Mostramos en la imagen la relación entre la memoria lógica y la física a través de la Tabla de Descriptores de Segmento (TDS).



# ¡GRACIAS!

---



*Ing. Francisco Yuman*



*1603760330101@ingenieria.usac.  
edu.gt*

