



UAEM

Universidad Autónoma  
del Estado de México



CENTRO UNIVERSITARIO UAEM ATLACOMULCO

ELIZABETH EVANGELISTA NAVA

SISTEMAS OPERATIVOS

PROTAFOLIO

ROCIO BELTRAN MONROY

ROCIO BELTRAN MONROY



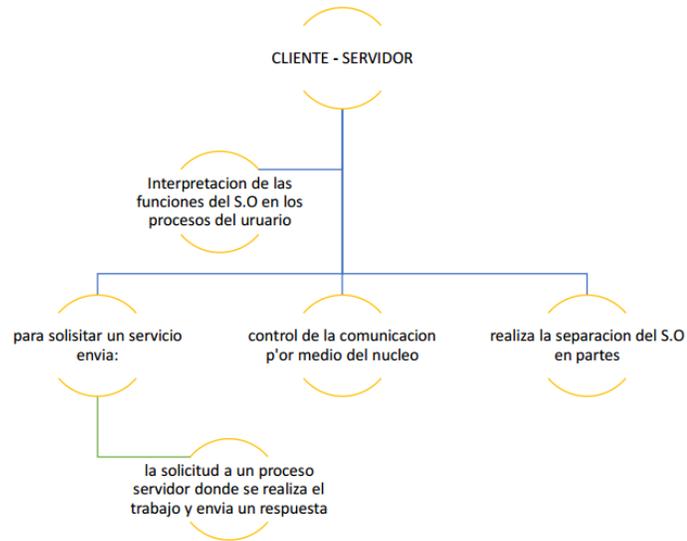
## GENERACION CARACTERISTICAS

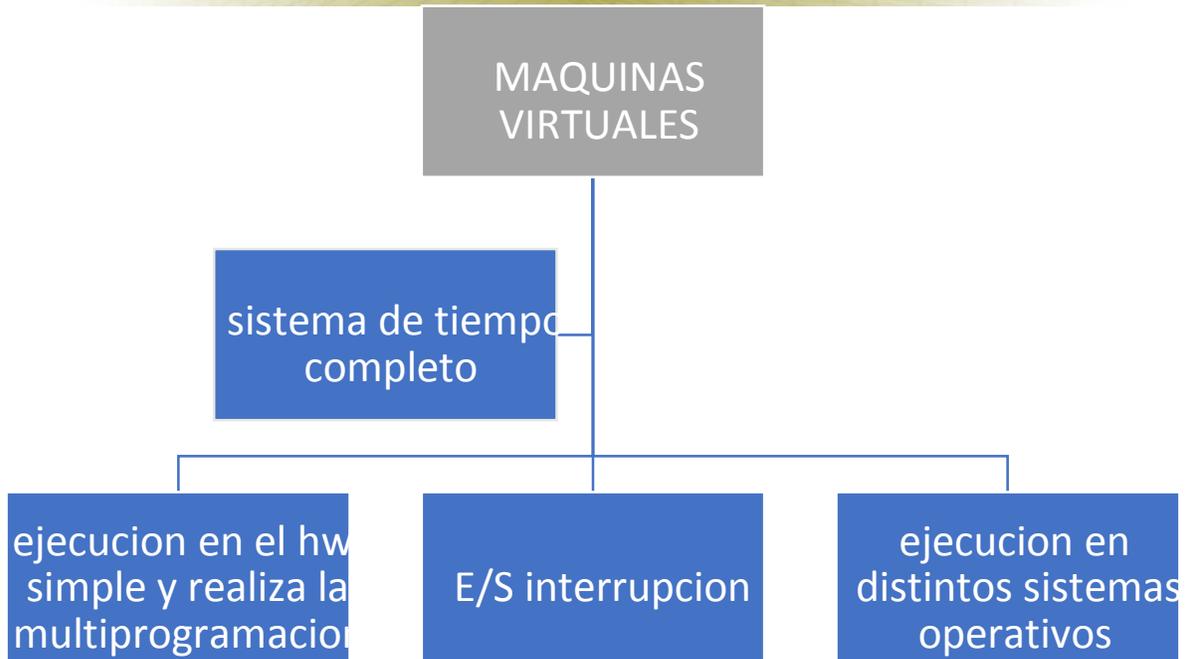
<i>Concepto de sistema operativo (década de los 50)</i>	Primer S.O fue creado en 1956 para un ordenador IBM 704, lo que hacía era comenzar la ejecución de un programa cuando el anterior terminaba.
<i>Revolución de los S.O (60's)</i>	Aparecen conceptos como sistema multitarea, sistema multiusuario, sistema multiprocesadores y sistema en tiempo real.  También aparece UNIX, que es la base de los s.o. actuales
<i>Ordenadores públicos (70's)</i>	Los ordenadores ahora eran personales y salieron al comercio. Se multiplica el desarrollo, creándose el lenguaje de programación C.
<i>Nuevos sistemas operativos (80's)</i>	MacOS, MS-DOS, Windows
<i>LINUX (90's)</i>	Se publica la primera versión del núcleo en septiembre de 1991, se unió al proyecto GNU, un sistema operativo libre, similar a UNIX, al que le faltaba para funcionar un núcleo funcional.
<i>Tubos al vacío 1945-1955</i>	Para que se pudiera programar se necesitaba de cables, interruptores y tarjetas perforadas
<i>Transistores 1955-1965</i>	Este tuvo una reducción de tamaño y precio, para la programación era de ensamblador por medio de tarjetas perforadas con un procesamiento por lotes.

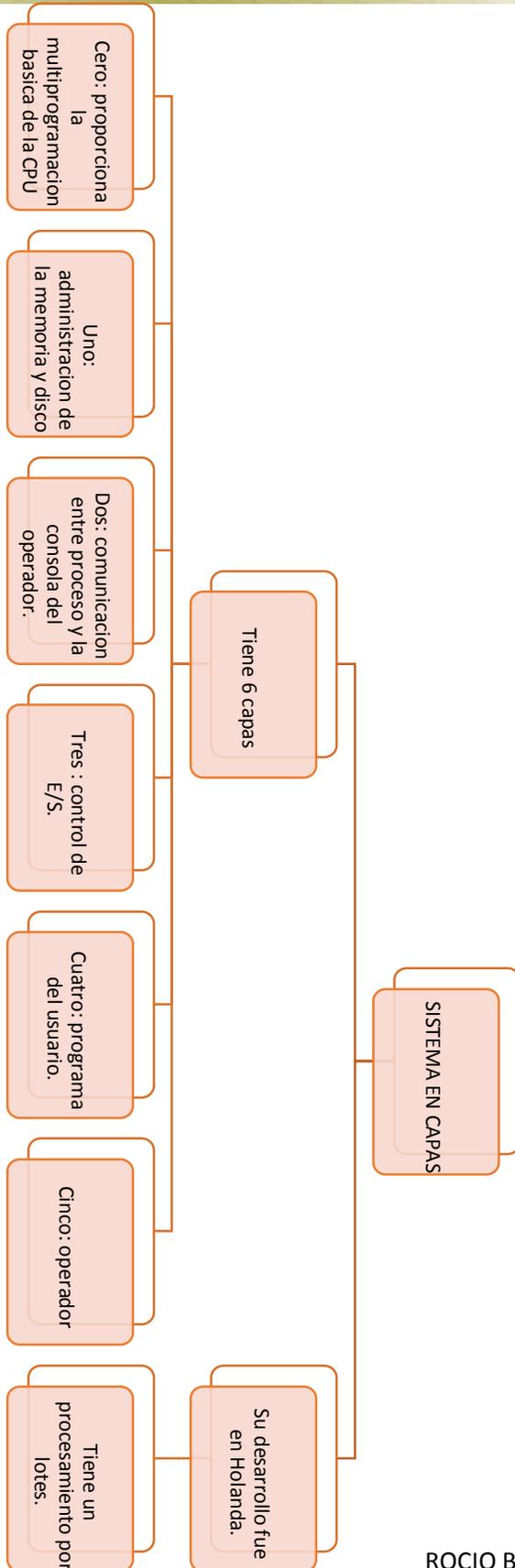


*Circuitos integrados 1965-1980* Multiprogramación, spooling, tiempo compartido.

ROCIO BELTRAN MONROY	12/FEB/2015
Cliente-Servidor	Tipos de S.O
ALCANCES: clasificación de los S.O por su estructura	











Sistema Operativo	Características	Ventajas	Desventajas	Tipo de núcleo	Shell
LINUX	Basado en UNIX Multitarea Multiusuario Sistema de código abierto Sistemas de red	Estable Rápido Libre No restricciones Seguro	Difícil de usar	Monolítico	Permite al usuario interactuar con el kernel con la interpretación de los comandos
UNIX	Multiusuario, capacidad de simular multiprocesamiento y procesamiento no interactivo. Escrito en un lenguaje de alto nivel : C.	Estable Diseñado en lenguaje de alto nivel. Portabilidad	Comandos especiales	monolítico	Provee al usuario una interfaz hacia el sistema operativo, el usuario dialoga con el intérprete de comandos.
WINDOWS	Lista de accesos directos. Compatibilidad Mejor rendimiento multitareas	Contiene muchas aplicaciones. Compatibilidad de controladores.	Limitaciones por RAM No soporta archivos nfs No bloqueo de intrusos	monolítico	<a href="#">interfaz de usuario</a> se presenta, incluyendo la <a href="#">barra de tareas</a> , el escritorio, cuadros de diálogo y controles de interfaz



MAC	Integración total con iCloud Seguridad mejorada Safari 6 Power nap	Interfaz amigables No tantos ataques de virus. Equipo confiable	Poca compatibilidad Menor catálogo de juegos Muy caro	hibrido	Lista de comandos
ANDROID	Conectividad Soporte de java Almacenamiento Multi-táctil Multitarea tetherin	Código abierto Más de 100.000 aplicaciones	Poca duración de pila Poco intuitivo Es totalmente fragmentado	monolítico	Programa que escucha a la entrada del teclado de un usuario y realiza acciones como lo indique el usuario.



UAEM

Universidad Autónoma  
del Estado de México



## INSTALACION DE CENTOS

### UNIDAD DE COMPETENCIA II “TIPOS DE SISTEMAS OPERATIVOS”

ROCIO BELTRAN MONROY

**OBJETIVO:** el alumno instalara el sistema operativo LINUX de la distribucion de CENTOS, con la finalidad de identificar y poner en practica el procedimiento de su implemetacion.

#### **ALCANCES**

- identificar de una manera generalizada los componentes de un sistema operativo.
- Analisis del proceso de instalacion
- Utilizar una maquina virtual

#### **REQUERIMIENTOS**

- 1GB de memoria
- 20gb de disco duro
- Puertos usb
- Unidad de cd
- S.O centos
- Virtual box

**DURACION:** 2hrs (4 hrs. estimado).

#### **INTRODUCCION**

La utilizacion de un S.O no solo permitira la gestion de recursos de forma individual, tambien permite el uso compartido de recursos que se encuentran interconectados a traves de una red.



UAEM

Universidad Autónoma  
del Estado de México

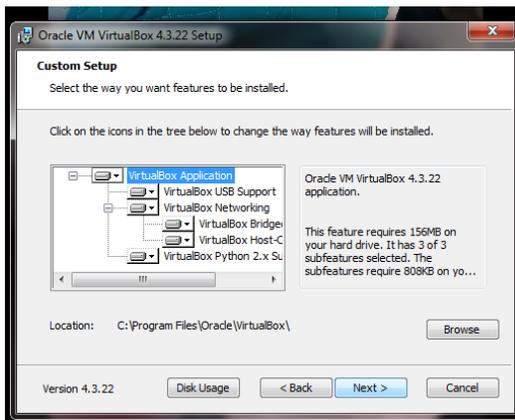


## PASOS A SEGUIR

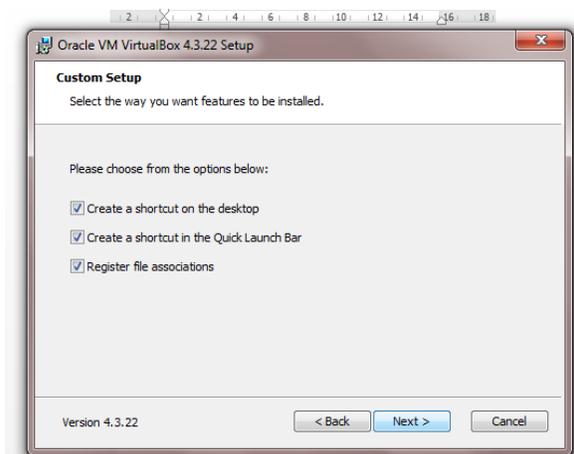


Se comienza la instalación de virtual box.

La pantalla nos da la bienvenida y nos dice que continuemos.



Nos indica que vamos a instalar virtual y continuamos



Tenemos que seleccionar que es lo se va a crear y se seleccionan todas

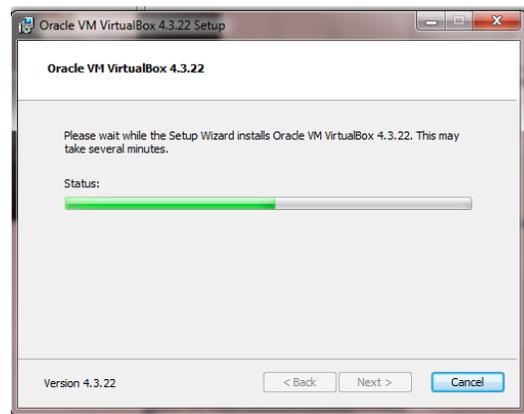
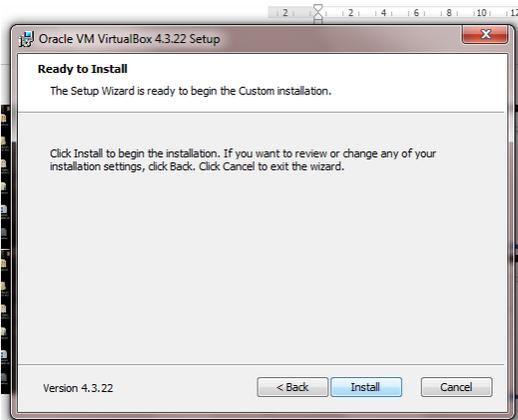


UAEM

Universidad Autónoma  
del Estado de México



Advertencia de creación de interfaz



En las tres pantallas anteriores se estaba instalando la aplicación de virtual y aceptamos al editor Oracle.

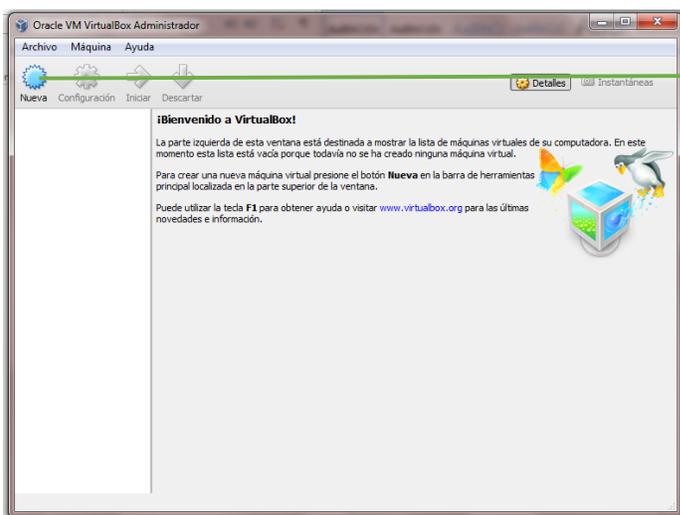


# UAEM

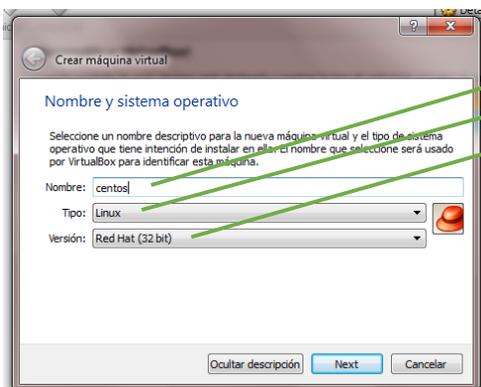
Universidad Autónoma  
del Estado de México



Bienvenida a virtual box



Aquí vamos a crear la máquina virtual

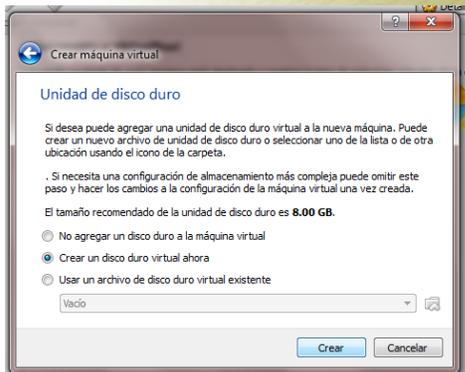


Nombramos a nuestra máquina,  
seleccionamos el tipo LINUX y la  
versión de red hat de (32 bit)

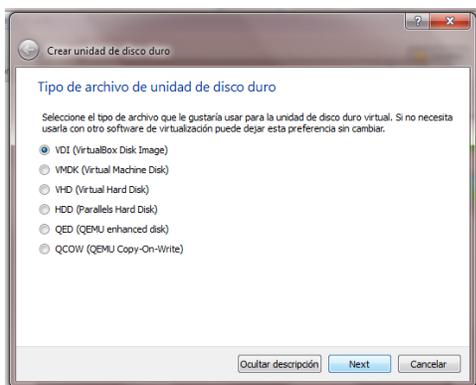


UAEM

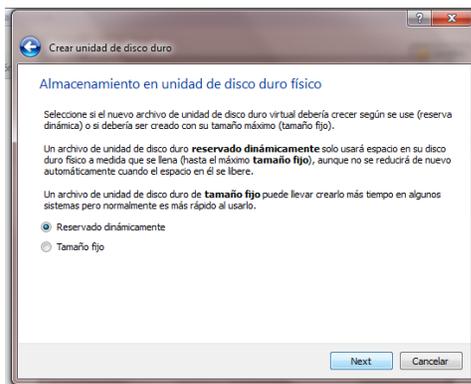
Universidad Autónoma  
del Estado de México



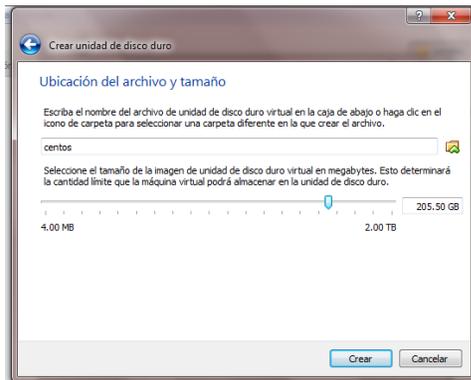
Selección de disco duro virtual



El tipo de archivo del disco es VDI



El almacenamiento es dinámico  
reservado

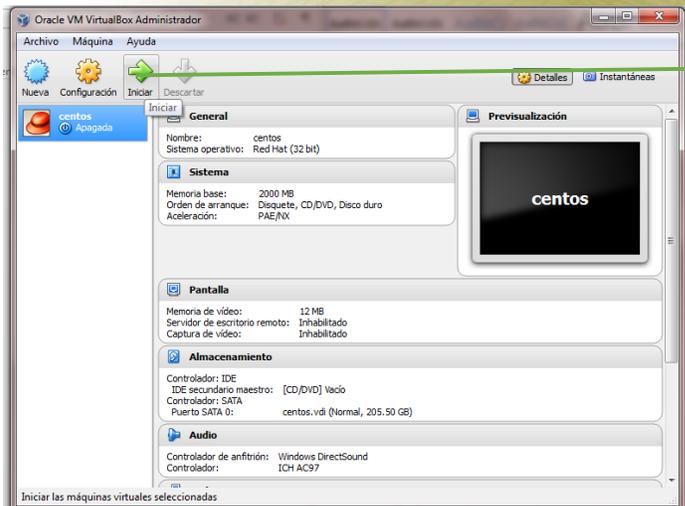


El tamaño de imagen es de 205.50 gb

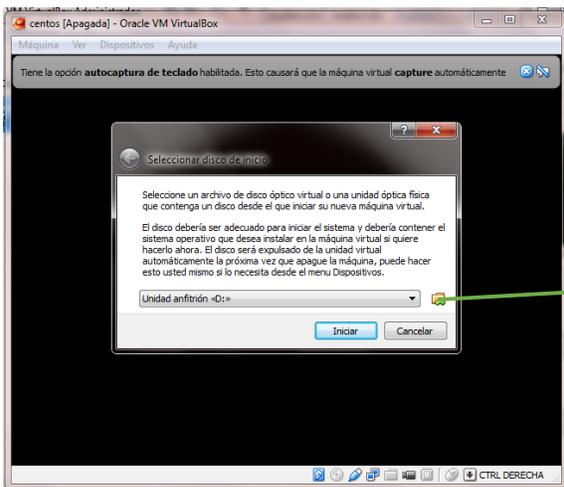


UAEM

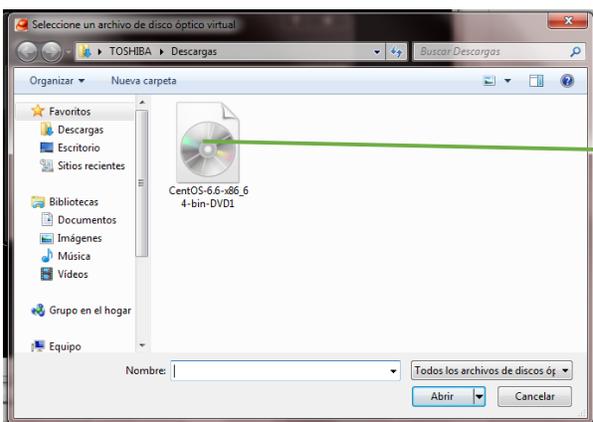
Universidad Autónoma  
del Estado de México



Iniciamos la instalación de centos  
(iniciamos nuestra máquina virtual)



Buscamos la unidad donde se  
encuentra centos

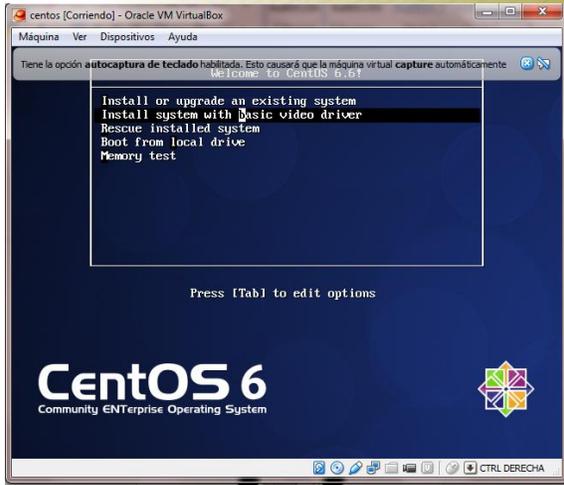


Ya que se encontró la unidad  
seleccionamos centos de 64 bits

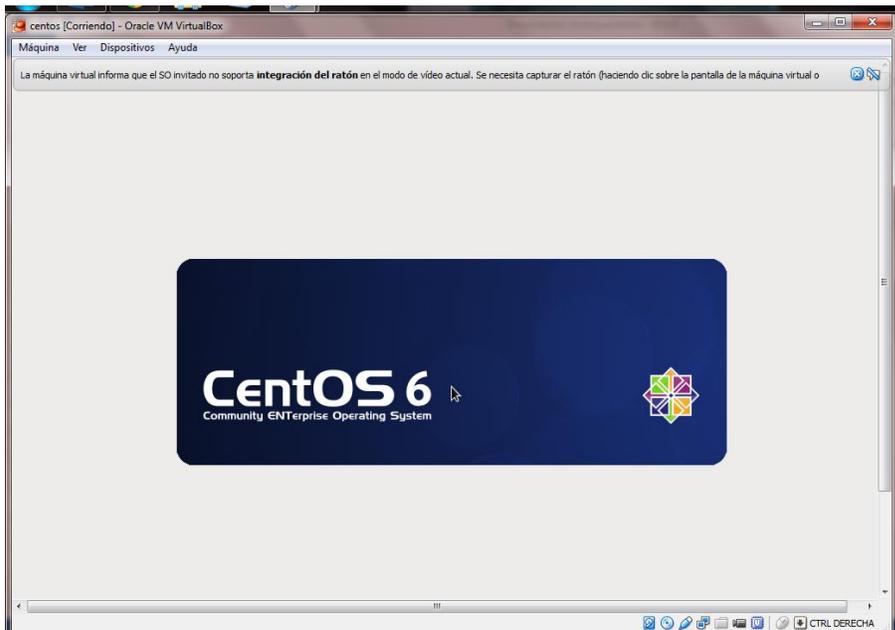


# UAEM

Universidad Autónoma  
del Estado de México



Una vez que lo seleccionamos comenzara el proceso de instalación. En esta pantalla elegimos instalación del sistema con disco

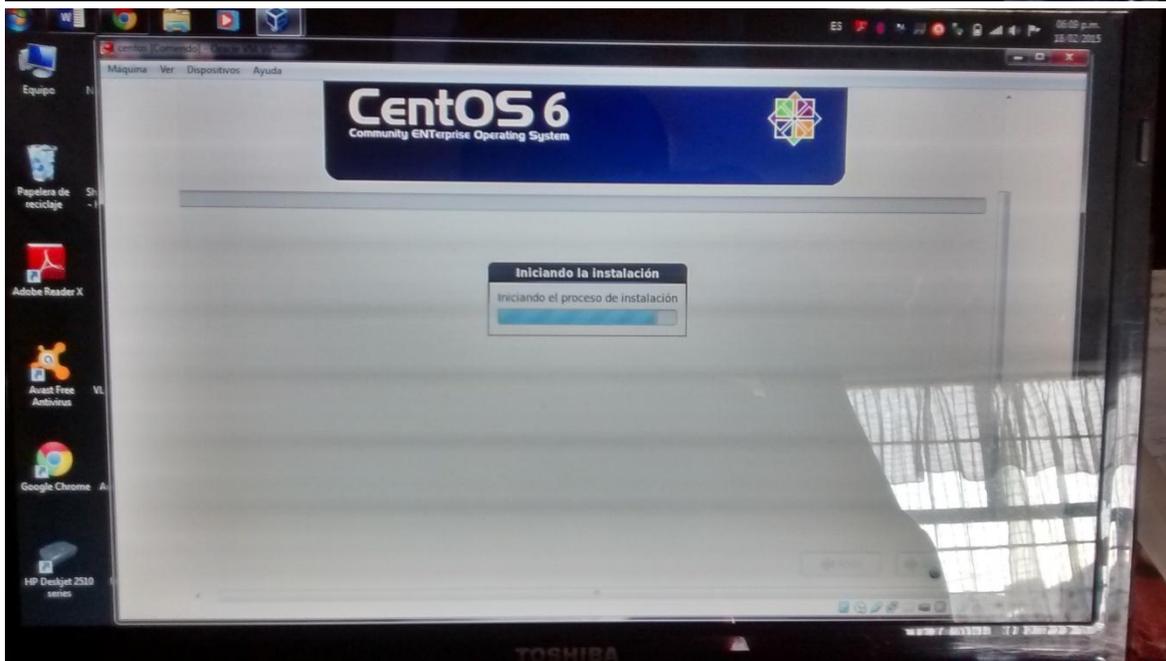
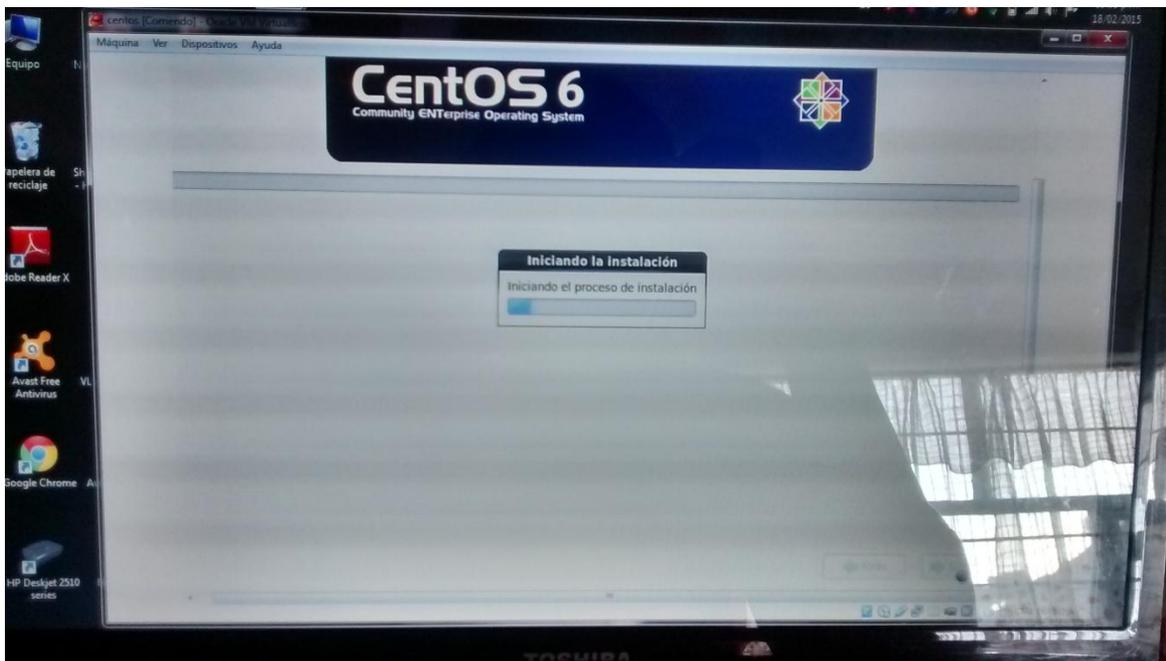


iniciamos



UAEM

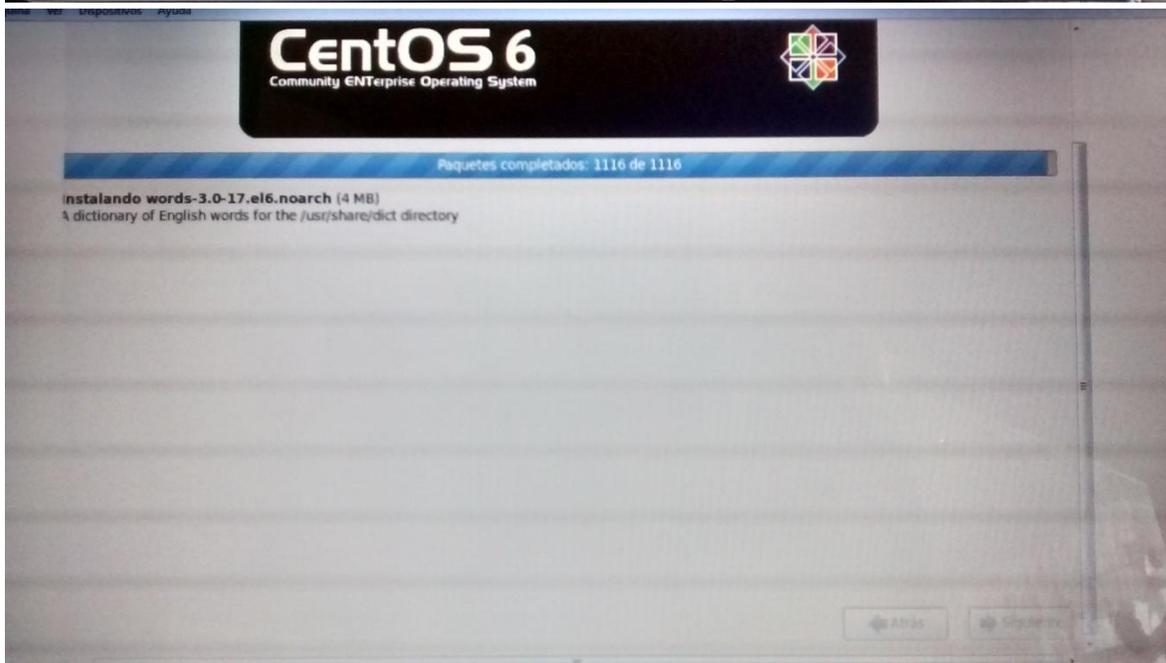
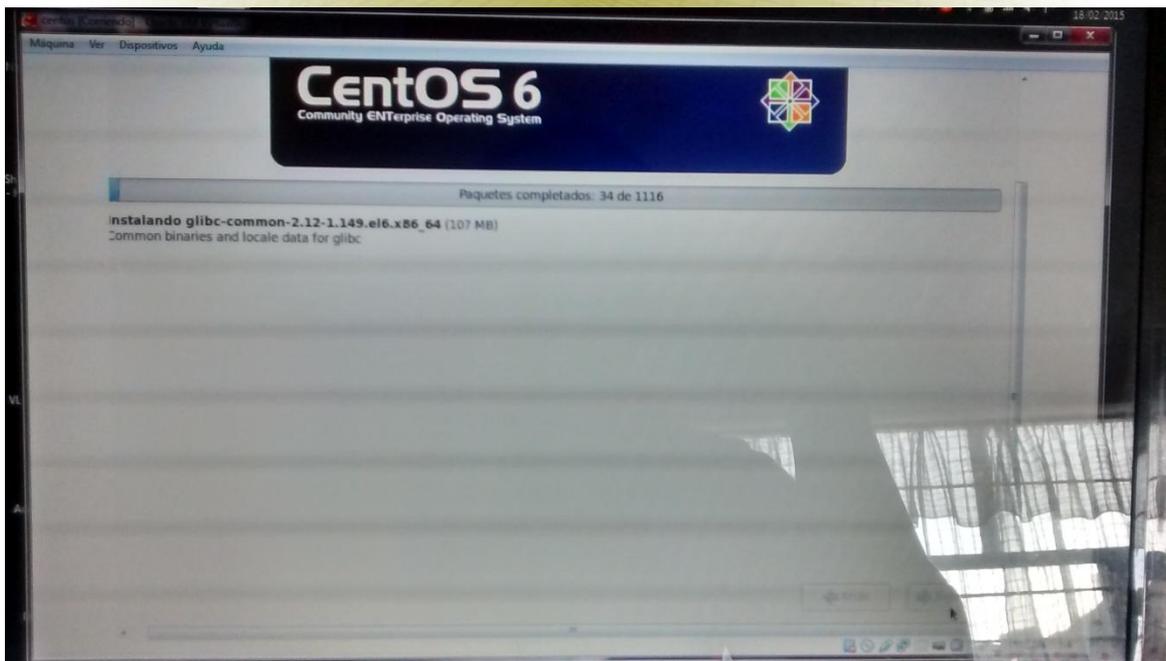
Universidad Autónoma  
del Estado de México





UAEM

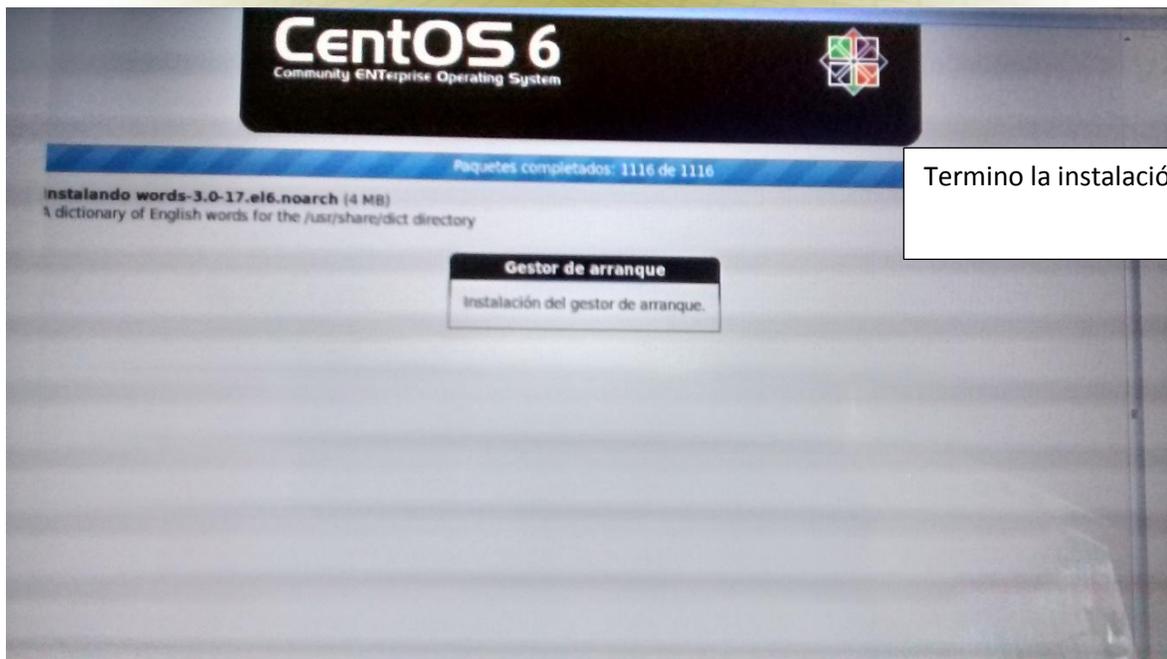
Universidad Autónoma  
del Estado de México



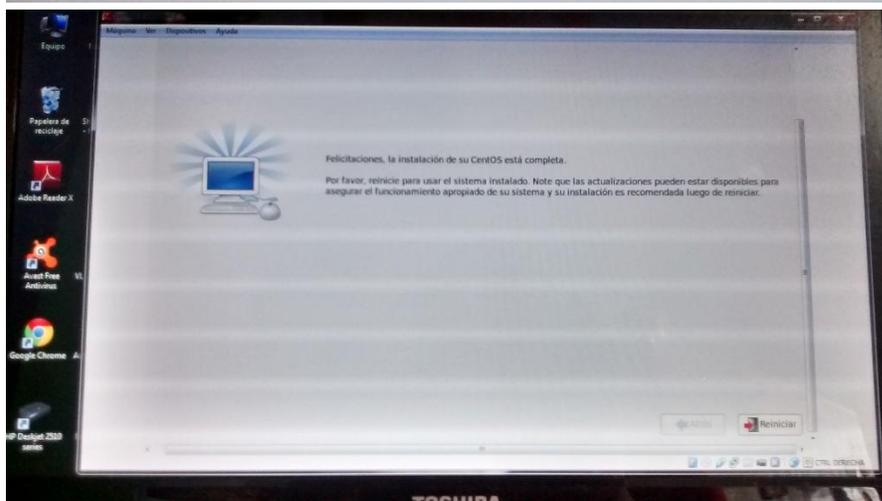


UAEM

Universidad Autónoma  
del Estado de México



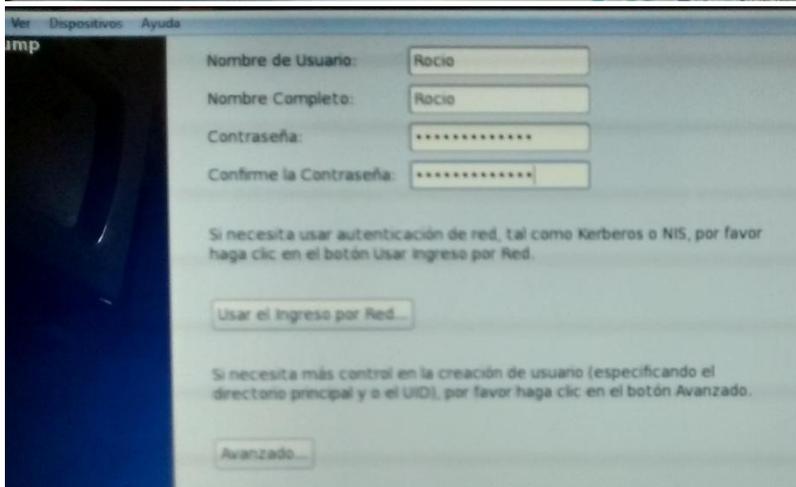
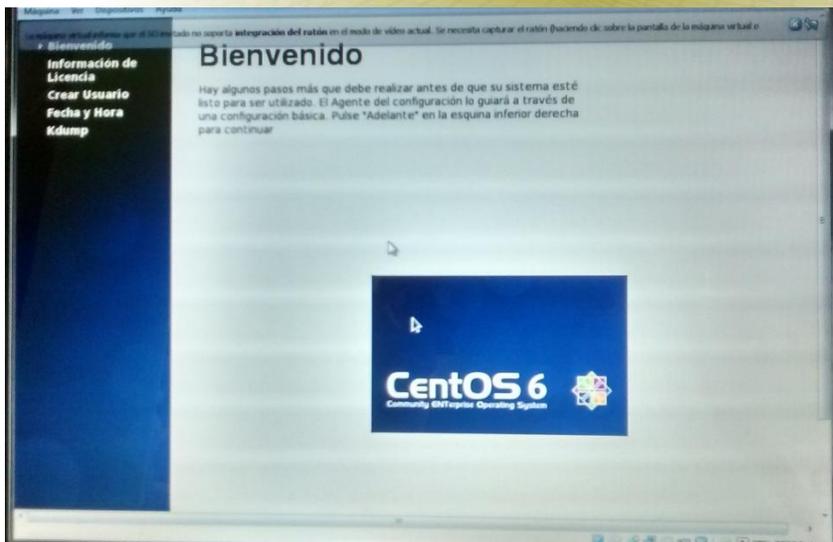
Termino la instalación de centos





# UAEM

Universidad Autónoma  
del Estado de México



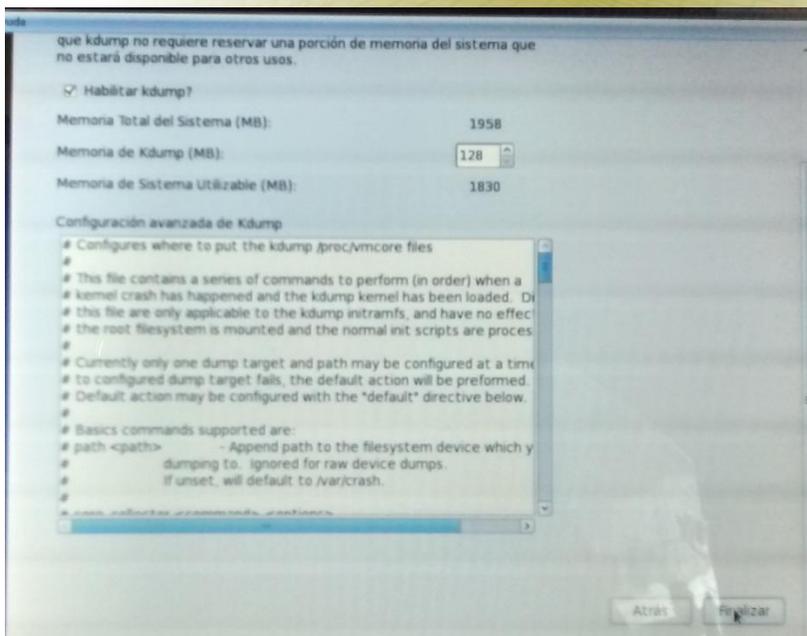
centos

Colocamos el nombre del usuario y se crea una contraseña.

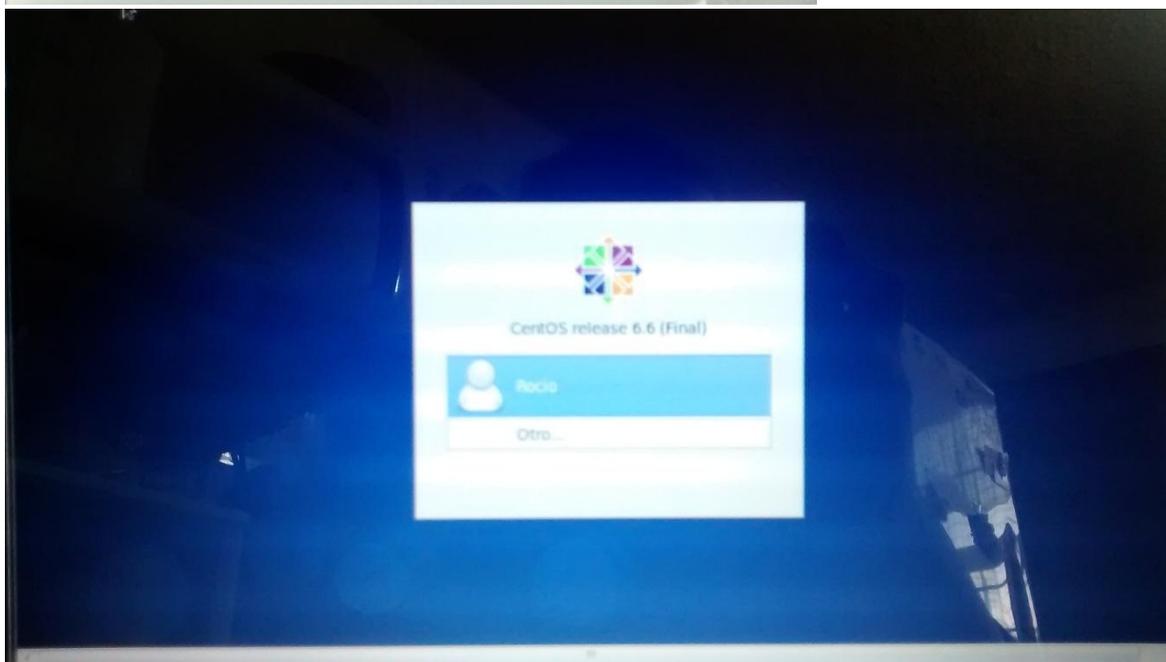


UAEM

Universidad Autónoma  
del Estado de México



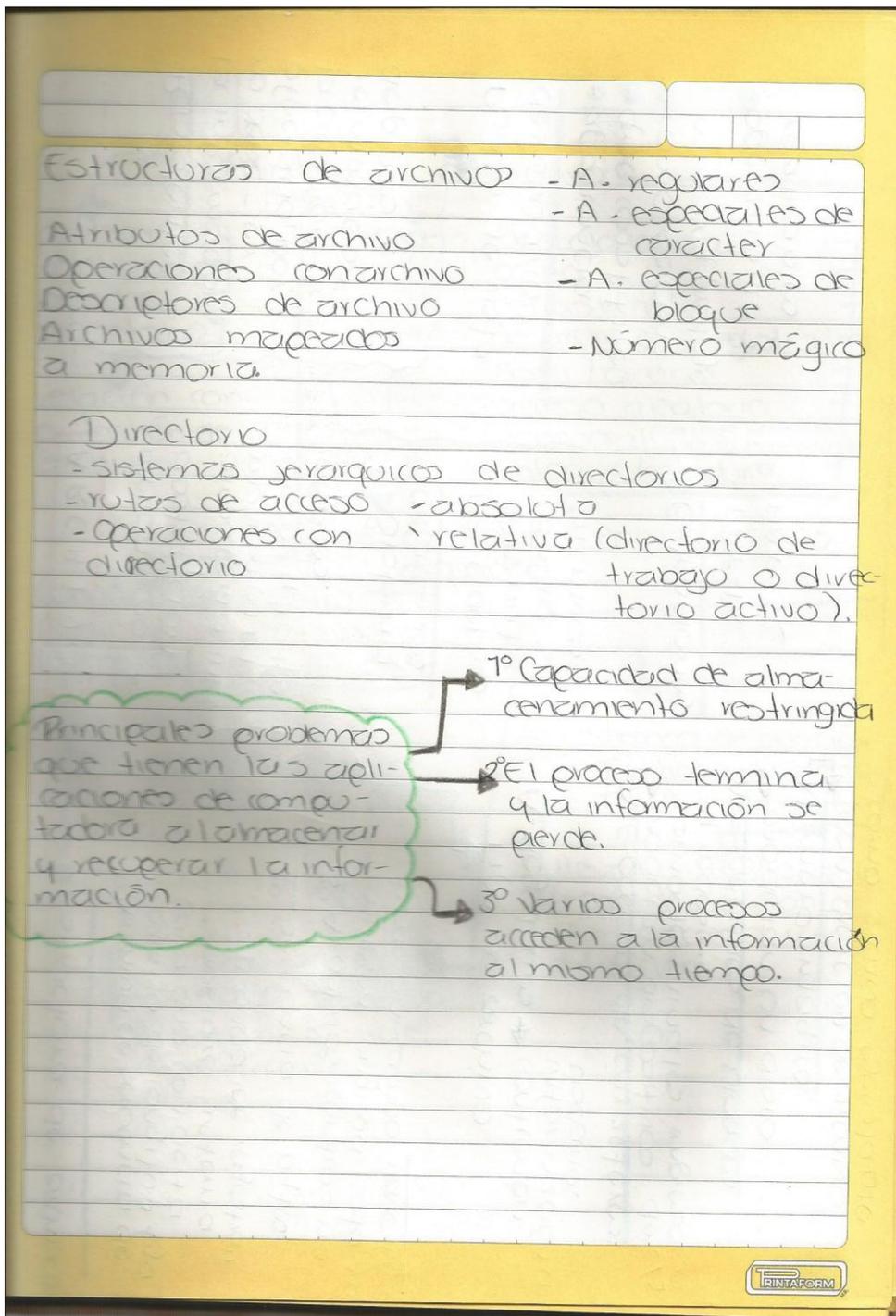
Habilitamos kdump

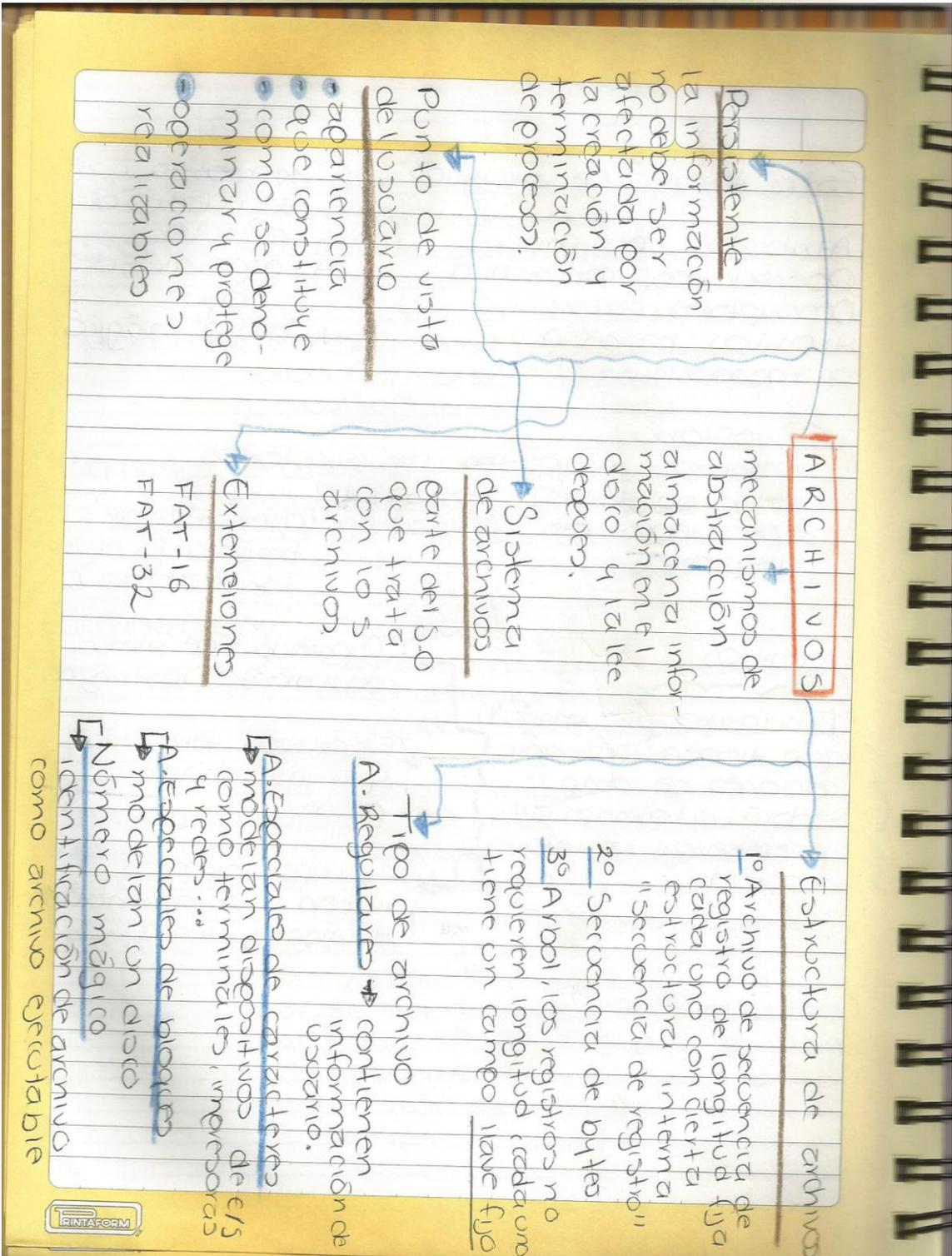


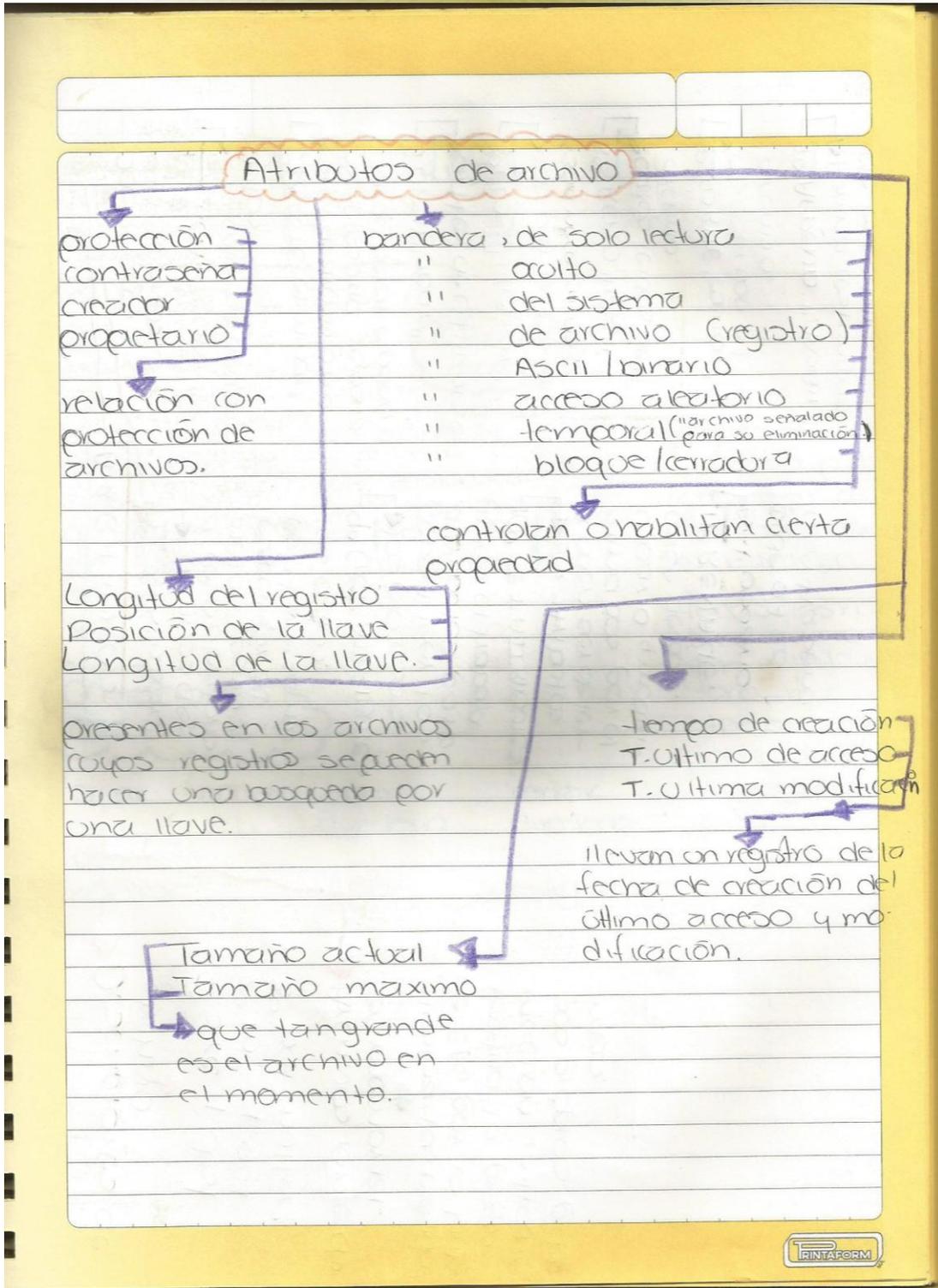
Nos indica nuestro usuario o nos da la  
opción de otro



Dentro de toda la instalación hubo muchos problemas a la hora de querer instalar centos ya que no se contaba con una buena versión. Pero al final salió todo bien y la instalación fue correcta









**OPERACIONES CON ARCHIVOS**

- 1º **CREATE "crear"** → el archivo se crea sin datos
- 2º **DELETE "eliminar"** → si el archivo no es necesario se borrar para liberar espacio
- 3º **OPEN "abrir"** → antes de utilizarse
- 4º **CLOSE "cerrar"** → los atributos no son necesarios.
- 5º **READ "leer"** → datos leídos del archivo.
- 6º **WRITE "escribir"** → describe el archivo
- 7º **APPEND "añadir"** → forma restringida de escribir

**ARCHIVOS**

- 8º **SEEK "buscar"** → archivos de acceso aleatorio método para especificar el punto de tomar los datos.
- 9º **GET ATTRIBUTES** → los procesos deben leer los atributos para hacer el trabajo
- 10º **SET ATTRIBUTES** → los atributos pueden ser determinados por el usuario y modificarse después.
- 11º **RENAME** → del usuario pide otro con trasfondo.

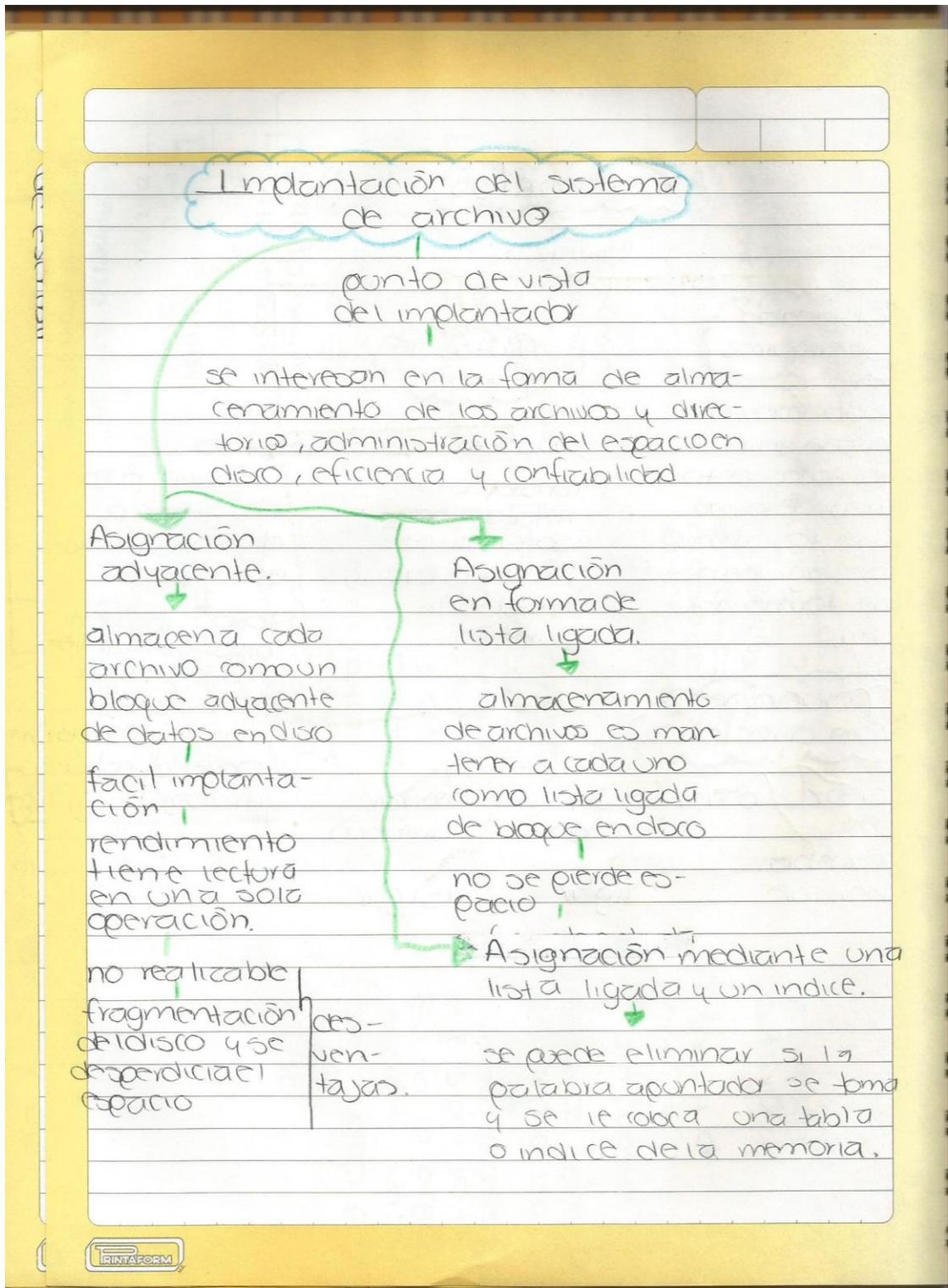
**Descripciones de archivo**

- 1º **src y dest**
- 2º **in y out**
- 3º **buf "buffer"**

**Archivos manejados en memoria**

- elimina la necesidad de E/S por lo tanto facilita la programación, introduce sus propios procedimientos.





**DIRECTORIOS**

- MS-DOS**

  - longitud de 32 bytes
  - atributos
  - nombre del archivo
  - numero del primer bloque
- UNIX**

  - los directorios pueden contener otros directorios, sistema jerarquico de archivos
- nodo-i**

  - tipos
  - tamano
  - tiempos
  - privilegios
  - bloques en disco
- CP/M**

  - solo existe un directorio lo que hace el sist. de archivos busca el nombre en un solo directorio.

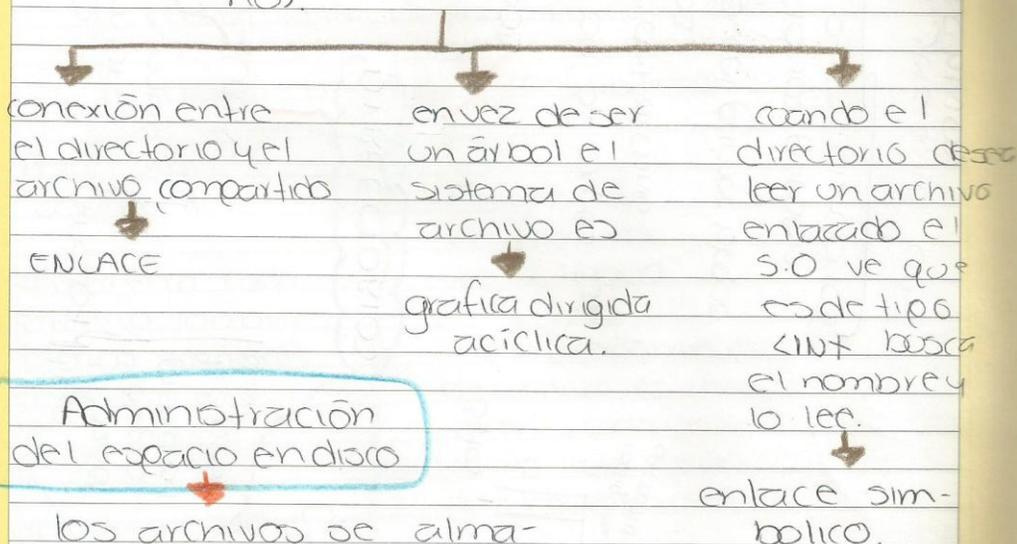
información relativa. (contenida en nodo-i)

copiador de directorios		punteros		número de bloques en disco	
1	nombre del archivo	3	tipo de archivo	2	contenedor
Bytes					
8		16		16	



### Archivos compartidos.

los archivos aparecen de forma simultánea en distintos directorios, los que pertenecen a distintos usuarios.



### Administración del espacio en disco

los archivos se almacenan por lo general en discos y es una preocupación fundamental

#### Estrategias

almacenar un archivo de  $n$  bytes

asignación consecutivos de espacio en el disco.





--	--	--	--

Tamaño de bloque  $\rightarrow$  fijo - medio en archivos  
 UNIX cercano a 1K

Administración del espacio en el disco.  $\rightarrow$  el administración del  
 Disk Quotas - asigna cada usuario una proporción un máximo de archivos

Confiabilidad del sistema de archivos.  $\rightarrow$  garantiza no exceder de sus cuotas

se puede adquirir un replazo "mínimo de alteraciones"  
 si se pierde será difícil restaurar la información o puede ser imposible.

no ofrece protección alguna contra la destrucción física del equipo y medios  
 "ayuda a la protección de la información"  
 Manejo de un bloque defectuoso  
 Respaldo  
 consistencia del sistema de archivos

Desempeño del sistema de archivos  $\rightarrow$  acceso a disco es mucho más lento que el acceso a la memoria

lectura de bloque de disco decenas de milisegundos, 100,000 veces más lento.  
 lectura de palabra de memoria tarda máximo centenas de nanosegundo





## Comandos de Linux

Objetivo: poder comprender el funcionamiento de los comando en la terminal de centos.

NOMBRE	FUNCIÓN	FORMATO
more	Permite desplegar el contenido de un archivo de texto a una plantilla a la vez	<COMANDO>   more
ls	Lista de contenido de un directorio, si se una sin un argumento este lista el contenido del directorio actual.	man ls
cd	Cambia el directorio corriente en que nos encontramos	cd
mkdir	Crea entradas de directorios	mkdir <nombre del directorio>
man	Proporciona información sobre los comandos u otros programas del entorno Linux	man man
rm	Elimina archivos o directorios	rm <nombre de los archivos> rm -r <nombre de los directorios>
exit	Cierra el Shell actual	
du	Reporta el espacio de disco utilizado por los archivos y directorios	du -ks <archivos o directorio>
eject	Extrae el cd-rom	
mv	Mueve los archivos o directorios	mv
cp	Copiar archivos. Se agrera un modificador "-r"	cp <archivo origen> <destino> cp -r <directorio origen> <destino>
passwd	Cambia la contraseña del usuario	passwd
cat	Concatenar y desplegar el contenido de archivo	cat
date	Muestra y establece la fecha y la hora	date
find	Buscar los archivo	find
hostname	Permite conocer la infomacion del equipo anfitrión	hostname



# UAEM

Universidad Autónoma  
del Estado de México



# UAEM

Universidad Autónoma  
del Estado de México



info	Muestra la información del sistema	Info
help	Despliega la ayuda para un comando de manual	help
clear	Limpia la pantalla de la terminal	clear
ln	Crean enlaces simbólicos entre archivos.	ln

## ROCIO BELTRAN MONROY

**PROPÓSITO:** El alumno pondrá en práctica algunos comandos adquiridos durante la sesión teórica, e instalará el jdk de java, posteriormente creará un archivo fuente y lo ejecutará.

### ALCANCES:

El alumno sabrá asignar permisos, crear un código fuente mediante un editor de textos, y ejecutarlo con JAVA

1. Define que es un código fuente

Son las instrucciones que le colocan a un software de programación, con el podemos realizar diferentes operaciones.

2. Menciona cual es el editor de texto que utilizaste

El que mas se utiliza es el vi

3. Cuál es el comando para asignar permisos de ejecución y cita su sintaxis

Chmod sintaxis chmod nombre del archivo

4. Cuál es la diferencia entre un compilador y un interprete

Que el interprete es la que puede hacer la llamadas al sistema decodificar y el compilador se encarga de traducir



UAEM

Universidad Autónoma  
del Estado de México



5. Como se llama el compilador de java javac
- 6.Cuál es la diferencia entre un archivo binario y un RPM

Que el archivo binario cuenta con una carpeta bin que es la que permite ejecutar os archivos, mientras los rpm solo pueden hacer verificaciones, instalar y desinstalar y solicitar programas

7. Como escribirías la instrucción para descomprimir un archivo llamado archivo.tar.gz.

```
tar xcvf archivo.tar.gz
```

8. Como escribirías la instrucción para eliminar un archivo llamadojdk-14-2.1.bin.

```
Mkdirjdk-14-2.1.bin
```

9. Como escribirías la instrucción para asignar permisos de ejecución al archivojdk-14-2.1

```
chmod 777 /jdk-14.2.1
```

- 10.Cuál es la instrucción para ejecutar un código fuente en java

```
Java Test
```



UAEM

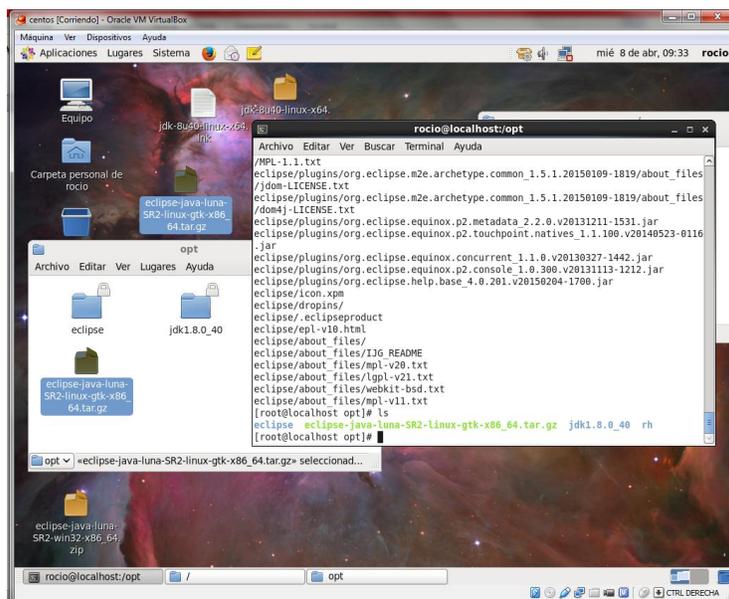
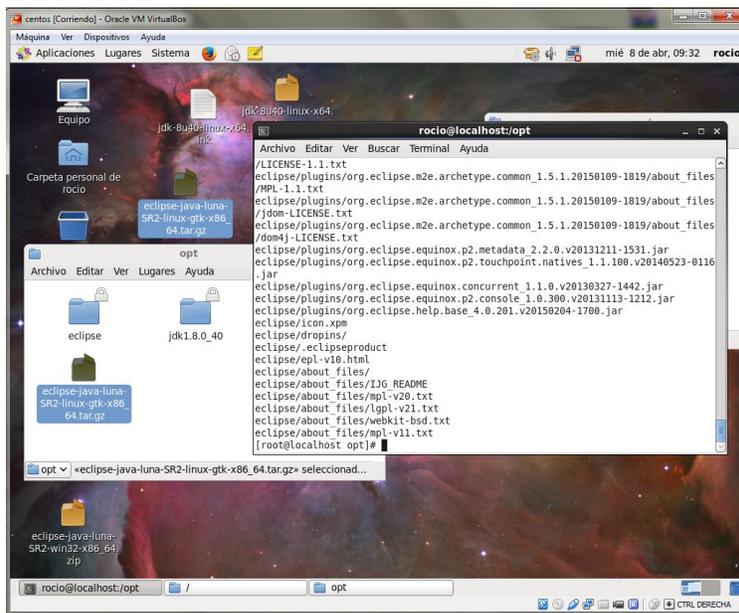
Universidad Autónoma  
del Estado de México



## PRACTICA

### Instalacion de eclipse en la maquina virtual

Se comienza la instalacion de elipse el programa se pasa al escritorio y despues se copia a la carpeta de OPT, se descomprime el programa para que este se pueda compilar.



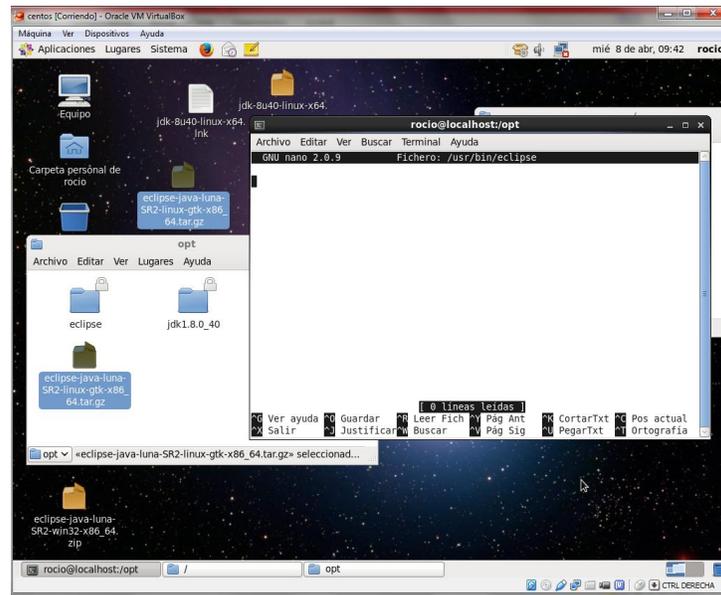


UAEM

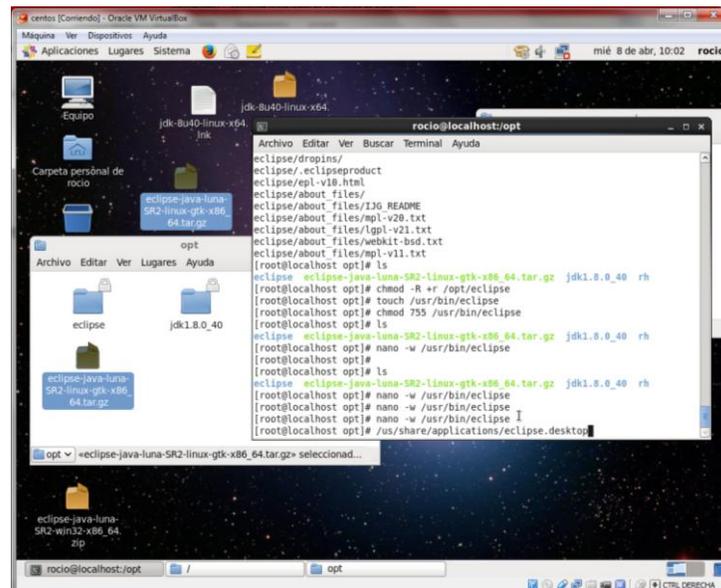
Universidad Autónoma  
del Estado de México



Después de que se descomprimió el eclipse se ejecuta donde damos la instrucción de crear un lanzador.



Se crea un archivo de texto donde se pone el código del lanzador (se nos pasó capturar la pantalla pero si se realizó la instrucción) después de crearlo nos fuimos a escritorio donde apareció un icono que dice eclipse y lo ejecutamos.

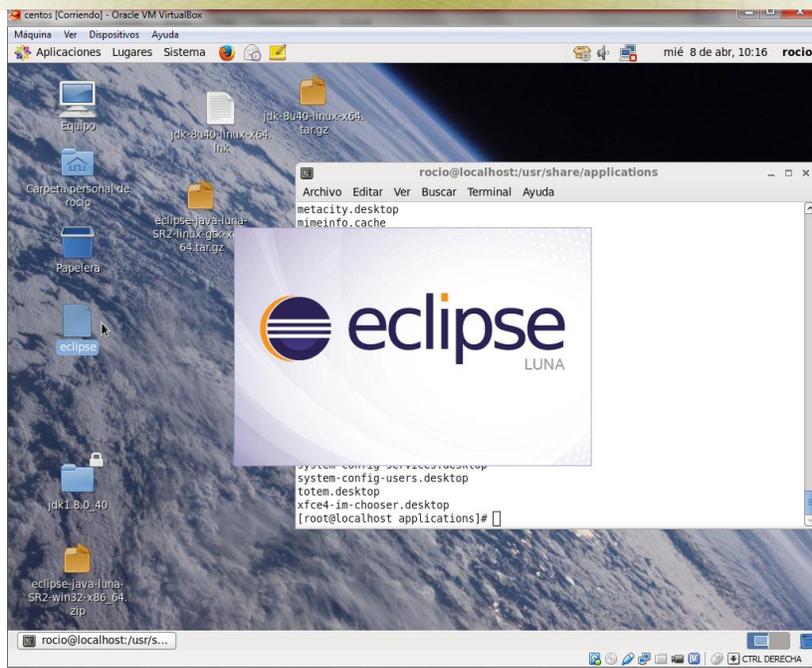


Comiza a ejecutarse el programa.

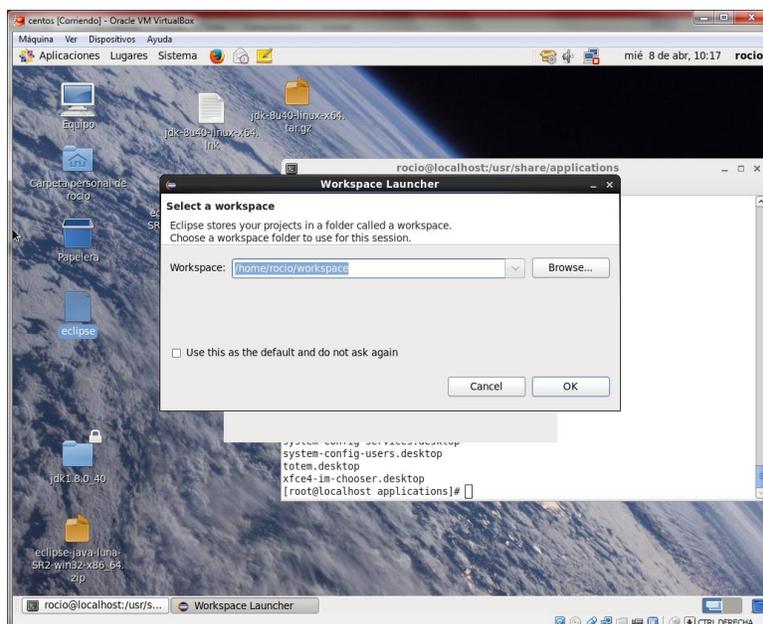


UAEM

Universidad Autónoma  
del Estado de México



Después de que se terminó de ejecutar nos aparece la siguiente pantalla del espacio de trabajo y le damos OK.

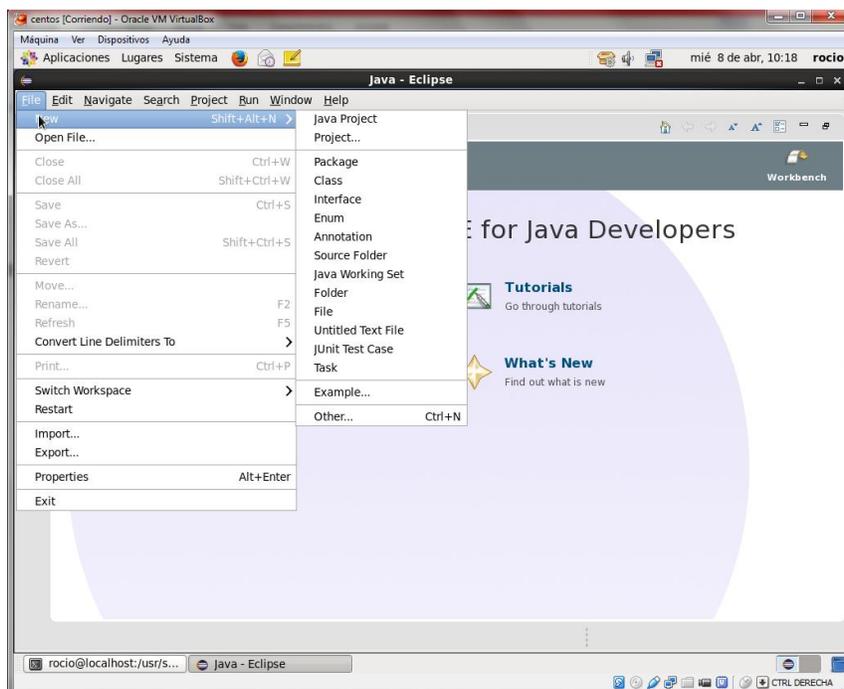
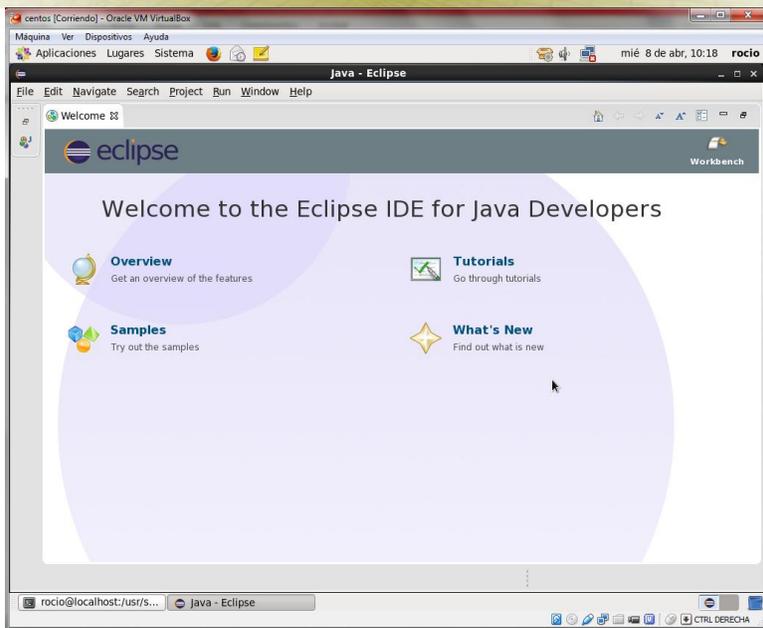


Ya nos encinramos dentro de eclipse



UAEM

Universidad Autónoma del Estado de México





UAEM

Universidad Autónoma del Estado de México

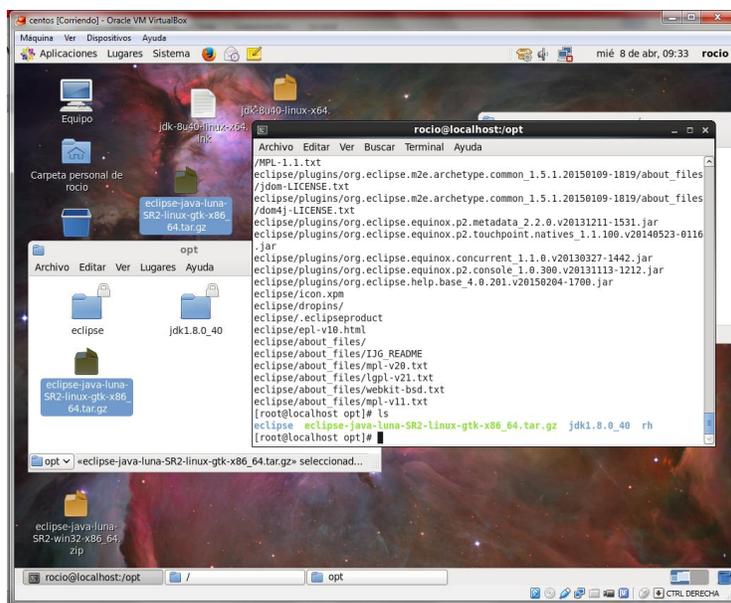
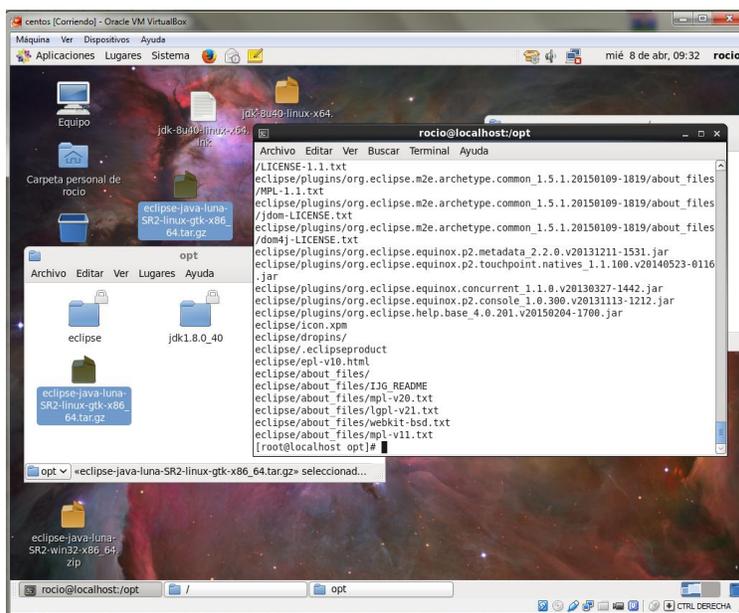


ROCIO BELTRAN

PRACTICA

Instalacion de eclipse en la maquina virtual

Se comienza la instalacion de elipse el programa se pasa al escritorio y despues se copia a la carpeta de OPT, se descomprime el programa para que este se pueda compilar.



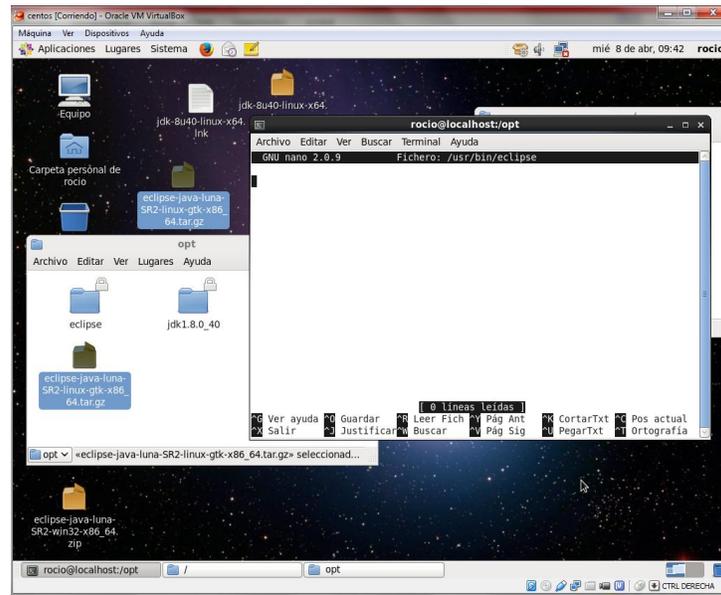


UAEM

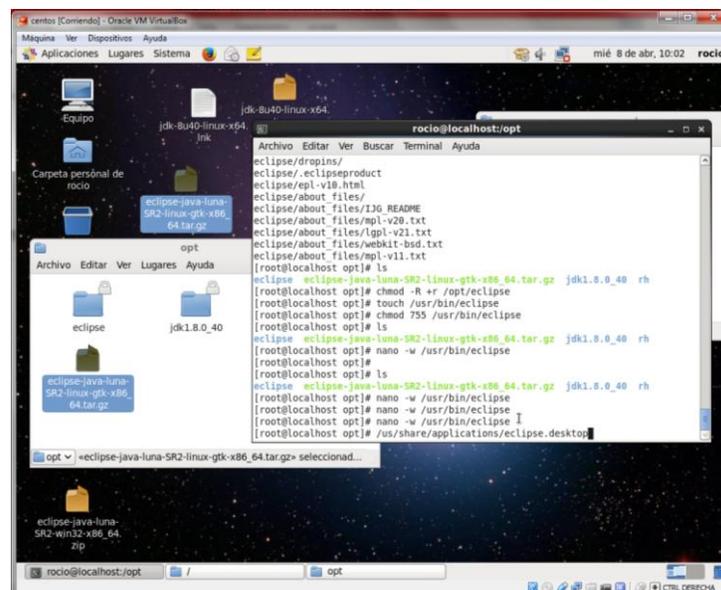
Universidad Autónoma del Estado de México



Después de que se descomprimió el eclipse se ejecuta donde damos la instrucción de crear un lanzador.



Se crea un archivo de texto donde se pone el código del lanzador (se nos pasó capturar la pantalla pero si se realizó la instrucción) después de crearlo nos fuimos a escritorio donde apareció un icono que dice eclipse y lo ejecutamos.



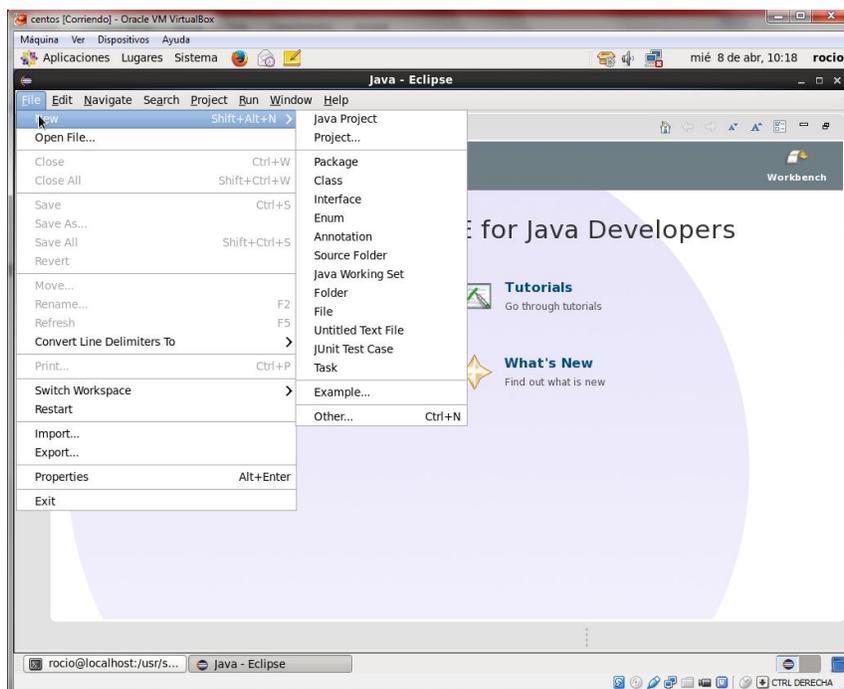
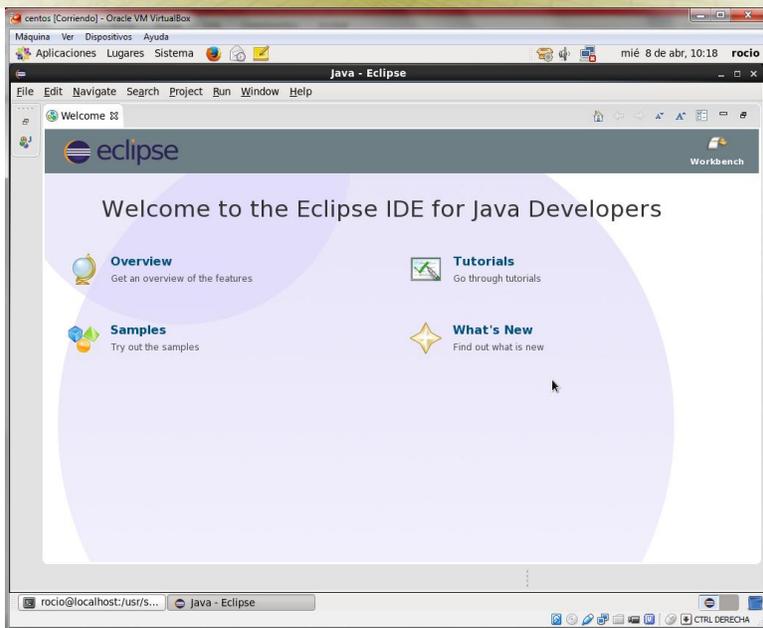
Comiza a ejecutarse el programa.





UAEM

Universidad Autónoma del Estado de México







```
#include "prototypes.h"
# define N5
void philosopher (int i)
{
while (TRUE
think ();
take_fork K(i);
take_fork (i+1), M);
```

Problemas de los lectores y de los escritores  
problema barbero, dormilon planificación de procesos.

El problema de los lectores y escritores  
↓  
Modela el acceso a una base de datos, sistema de reservas, varios procesos leen la base de datos al mismo tiempo  
↓  
Problema del barbero dormilon.  
La peluquería tiene un barbero una silla de peluquería y n de sillas para que se sienten los clientes en espera, si no hay.

Planificador - Algoritmo de planificación

- Equidad: garantizar que cada proceso obtiene una porción justa de la CPU
- Eficacia: mantener ocupada la CPU al 100% del tiempo.
- Tiempo de respuesta: minimizar el tiempo de respuesta para usuarios interactivos.
- Tiempo de regreso: minimizar el tiempo que el sistema espera los usuarios para obtener sus resultados
- Rendimiento: maximizar el número de tareas procesadas por hora.



Si no hay clientes presentes el barbero se sienta en su silla de peluquero y se duerme. Cuando llega un cliente se debe despertar al barbero oírmiton.

Planificación round robin  
Cada proceso tiene asignado un intervalo de tiempo de ejecución llamado quantum.

Proceso activo Sig. pág

Lista de procesos ejecutables

Lista de procesos ejecutables después de agotarse el quantum.



UAEM

Universidad Autónoma  
del Estado de México



## Unidad de competencia 5

Propósito: el alumno conceptualizara todos los fundamentos relacionados con procesos de estados de un proceso.

Comunicación entre proceso, problemas clásicos de planificación de proceso, planificación de procesos ara establecer relaciones de conceptos, y tomar en cuenta todos estos elementos durante el diseño de un sistema operativo.

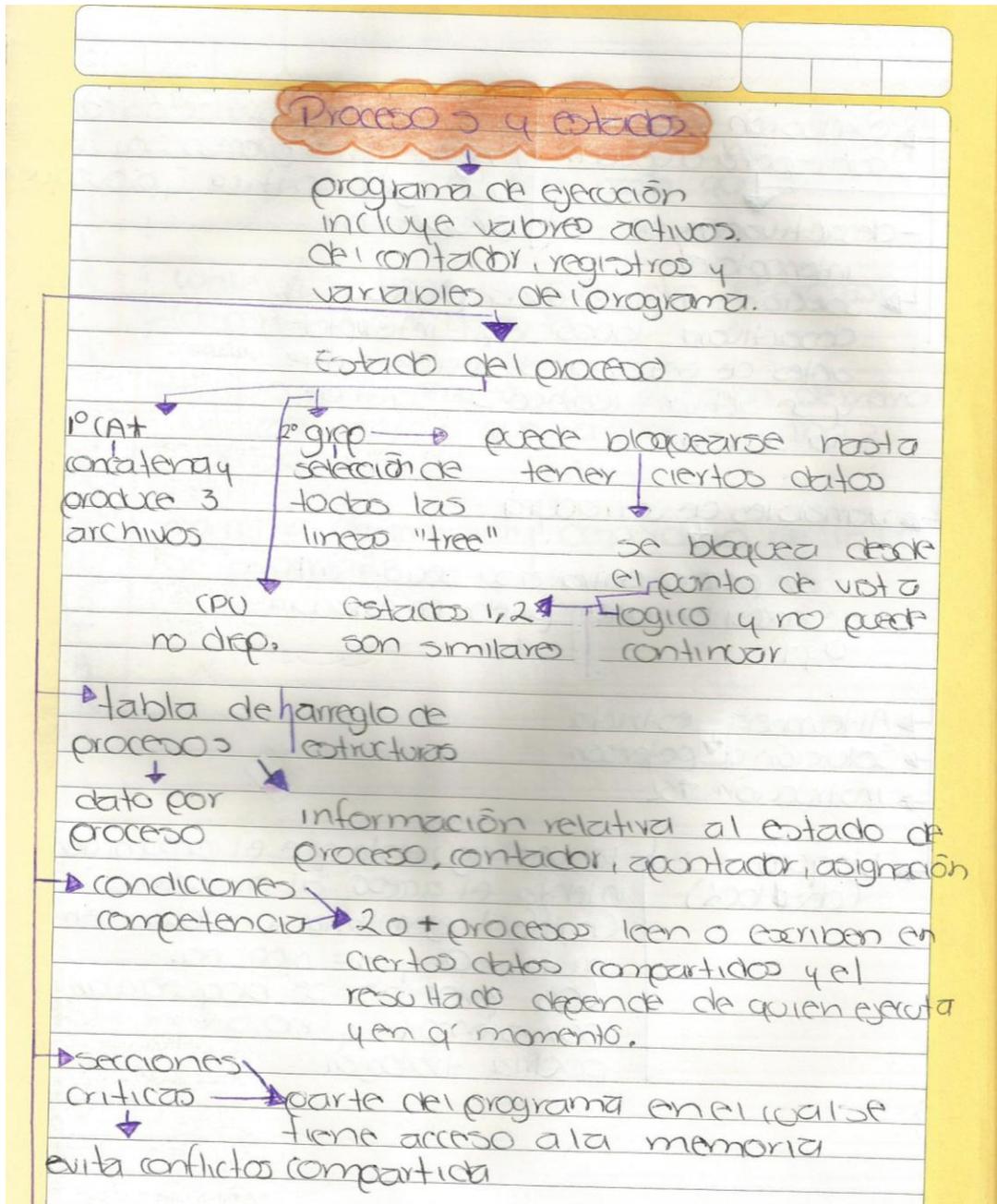
Alcances: conoce, identifica, compara y diferencia.

- Que es un proceso
- Estados de un proceso
- Problemas de comunicación
- Planificación

## Productos

Mapa concepto de procesos y estados

- conceptos tabla de procesos, condición de competencia, sección crítica, exclusión mutua, bloque (dead lock).
- Programa dormir despertar
- Programa productor – consumidor (puntos 2,3 exclusión mutua)
- Programa cena de filósofos
- Programa barbero dormilón (puntos 4,5 problemas de comunicación)
- Cuadro comparativo de planificación de procesos





→ exclusión mutua con espera ocupada. | si un proceso está ocupado ningún otro proceso entre su región crítica y provoca que problemas

↓

desactivación de interruptores.

→ solución simple cada proceso desactiva todos los interruptores antes de entrar a su sección crítica y se activan cuando se sale de la sección.

→ Variables de cerradura

↓

si un proceso entra a su sección crítica se necesita una prueba de cerradura 0 y 1.

→ Atención estricta

→ Solución de Peterson

→ Instrucción TSL

→ bloques (dead lock) | la vez siguiente que el consumidor intenta el acceso al almacén (buffer) ejecutaría un down en  $\text{mutex} = 0$  y se bloquea. Los procesos se bloquean por siempre y no se podría trabajar

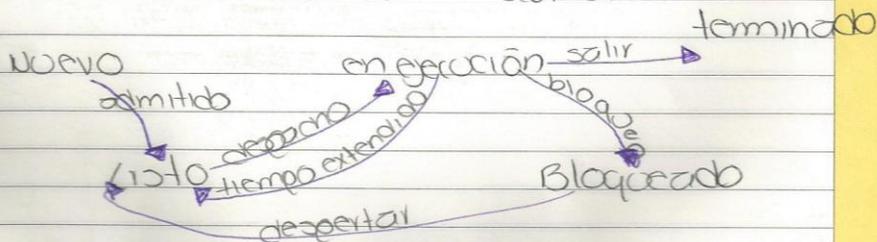


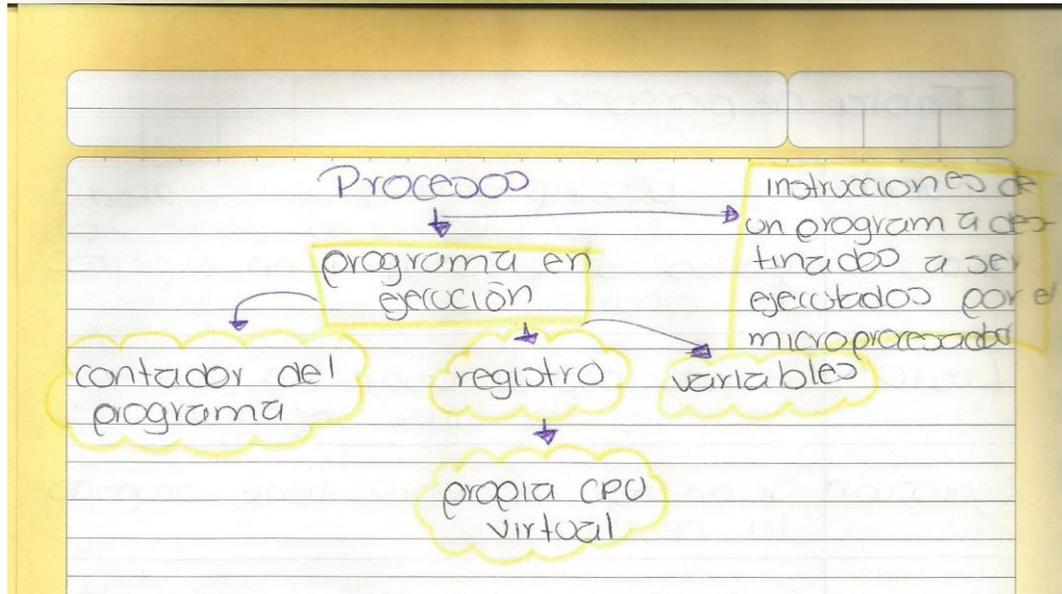
## Tabla de procesos

Estado	Descripción
Nuevo	Apenas se encuentra en el proceso de crearse.
Listo	Cuando podría usar un ucp si A existiera una disponible
Ejecución	Si en ese momento tiene ocupado la CPU
Bloqueado	Espera que ocurra algo, ejemplo to terminación de un EIS para poder ponerse en marcha
Terminado	Cuando un proceso se ha completado su ejecución para a ser un proceso terminado.

## Estado de procesos

el estado de proceso define el estado actual

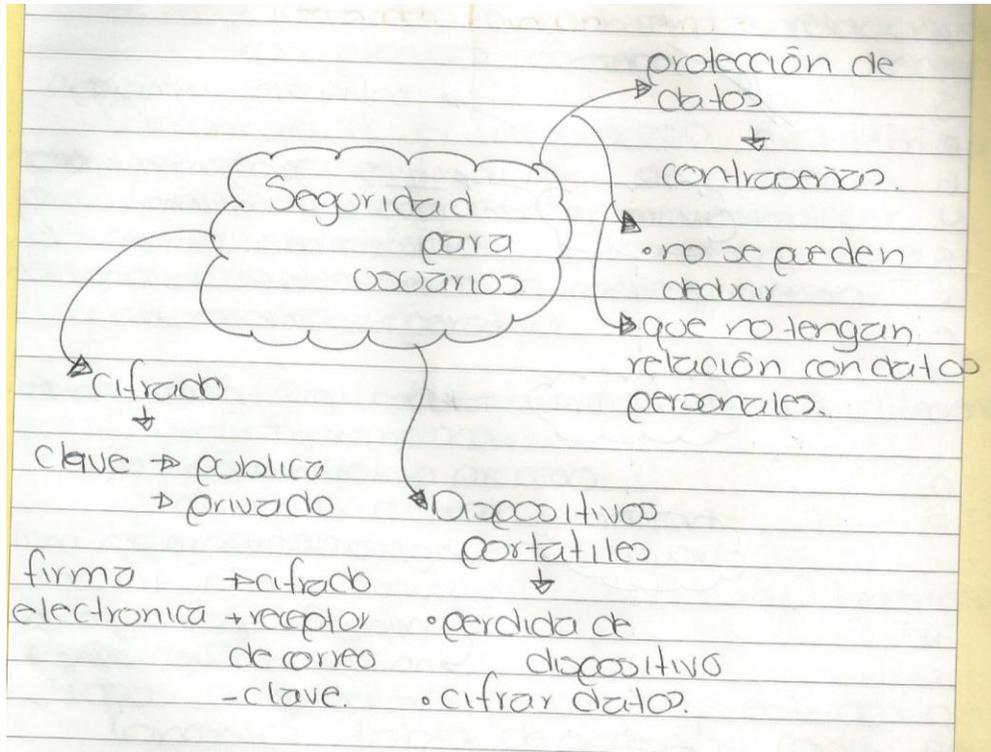


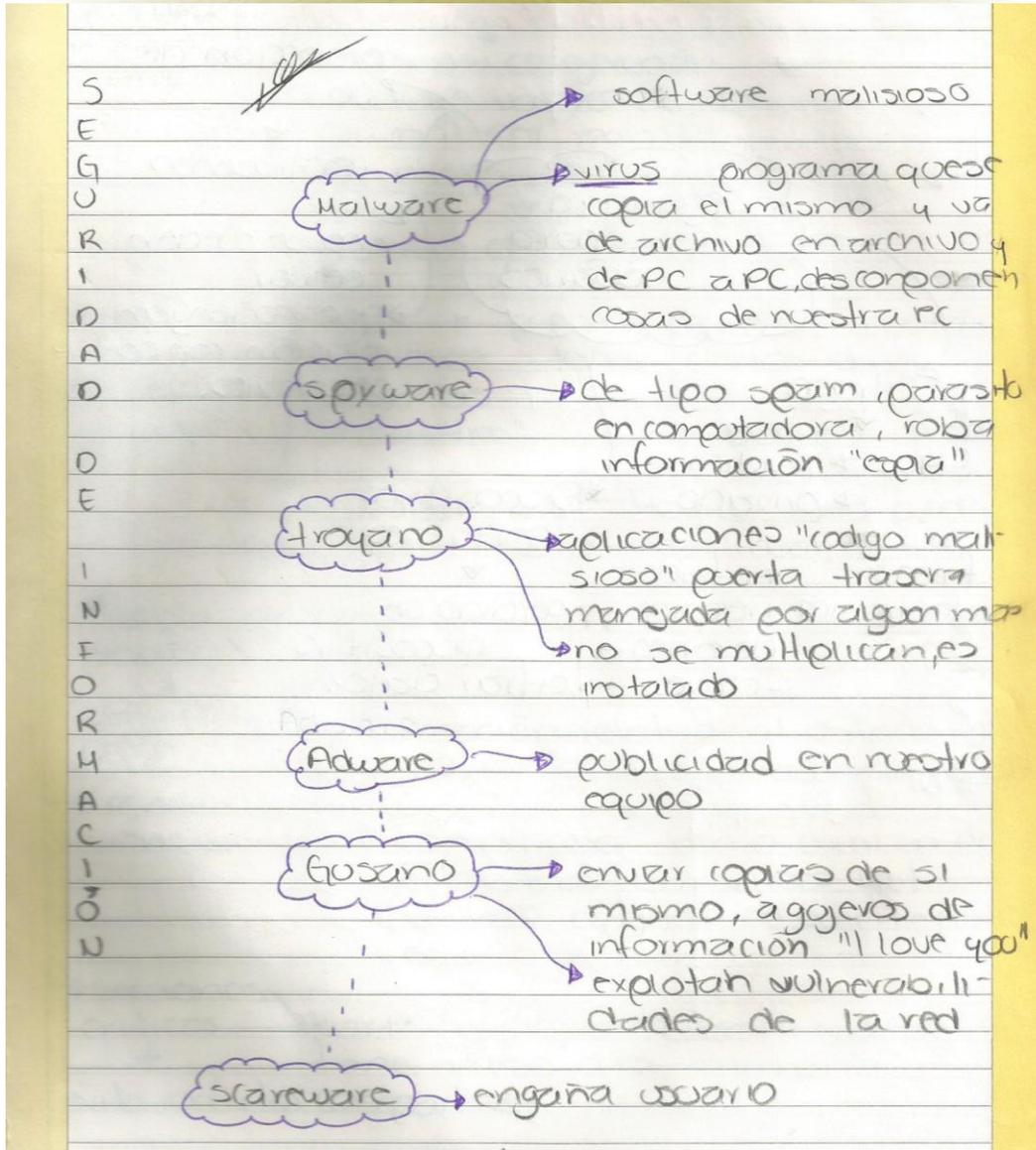




## SEGURIDAD DE LA INFORMACION

### Audios







Problemas entre procesos

Proceso	Problema	Solución.
Round Robin	- si el proceso continúa con la ejecución al final de cada proceso de su quantum, otro proceso tiene un interruptor de apropiación de tiempo de CPU, si el proceso "quantum" está bloqueado o terminado antes de consumir su quantum, se alterna el uso de la CPU. Aspecto interesante de round robin longitud del quantum. la alternancia entre un proceso y otro necesita tiempo para administración	- si el quantum es muy corto se alterna demasiado procesos, lo que reduce la eficiencia del CPU, pero si es muy largo esto puede causar una respuesta lenta a las solicitudes interactivas breves.
Por prioridad	Cada proceso tiene asignada una prioridad y el proceso ejecutable con máxima prioridad es el que tiene permiso de ejecución	Para evitar que los procesos de alta prioridad se ejecuten en forma refinida, como el planificador puede disminuir la prioridad del proceso en ejecución en cada instante (en cada interrupción del reloj). si esto hace que su prioridad sea menor que del siguiente proceso con alta prioridad, se alternan los procesos.



Colas múltiples	CTSS tenía el problema de que la alternancia entre procesos era muy lenta puesto que la 7094 subprograma mantenían un proceso dentro de la memoria representados en intercambio, el envío del proceso activo al disco y la lectura en el disco de un nuevo proceso.	Lo eficaz es de dar de una vez a los procesos con limitaciones de CPU un quantum más grande que darles pequeños quantum con frecuencia para reducir el inter-cambio. Al avanzar en las colas de prioridad, el proceso se ejecuta cada vez con menor frecuencia, lo que representa un ahorro de tiempo de CPU para los procesos interactivos cortos.
Primero e) trabajo más corto	tareas por lotes, cuando varios trabajos de igual importancia esperan en la lista de entrada para iniciar.	El algoritmo de primeros el trabajo más corto produce el promedio mínimo de tiempo de respuesta.
Planificación garantizada	Si existen $n$ usuarios dentro del sistema mientras $s$ se está trabajando recibirá cerca de uno entre $n$ de la potencia de la CPU.	El sistema debe tener un registro del tiempo de CPU que cada usuario ha tenido desde su entrada al sistema así como el tiempo transcurrido desde esa entrada. Calcula la cantidad del tiempo en CPU prometida al usuario el tiempo transcurrido desde la entrada.



*Beltrán*

Política vs mecanismo	Ninguno de los planificadores anticipados acepta los costos de los procesos del usuario relativos a decisiones de planificación, como resultado, el planificador hace la mejor elección	dividido entre el Separar el mecanismo de planificación de la política de planificación. El algoritmo de planificación que está parametrizado de alguna manera, pero los parámetros pueden ser determinados por medio de proceso del usuario
Planificación de dos niveles	- todos los procesos ejecutables se encuentran en la memoria principal, no se dispone suficiente memoria principal algunos procesos ejecutables se mantienen en el disco, el tiempo de alternancia entre procesos para tratar procesar una parte del disco es considerablemente mayor que el tiempo para un proceso que ya se encuentra en la memoria.	Un planificador de dos niveles debe separar los procesos entre el disco y la memoria, además debe elegir los procesos por ejecutar de entre aquellos que se encuentran en la memoria.



UAEM

Universidad Autónoma  
del Estado de México



## Practica en java de menú "el barbero dormilón"

```
1 package miproyecto;
2
3 public class menubarbero {
4
5 public static void main(String[] args){
6     System.out.println("problema clasico de sistemas operativos");
7     System.out.println(" el barbero dormilon");
8     System.out.println("menu");
9     int opcion;
10    switch(opcion){
11    case 1:
12        System.out.println("1.-El barbero esta dormido por que es un webon");
13        break;
14    case 2:
15        System.out.println("2.-Entra cliente");
16        break;
17    case 3:
18        System.out.println("3.-Sale cliente");
19        break;
20    case 4:
21        System.out.println("4.-Esta lleno");
22        break;
23    case 5:
24        System.out.println("5.-Salir del sistema");
25        break;
26    case 6:
27        System.out.println("6.-Elige una opcion");
28        break;}
29
30 }
```

```
31 }
32     System.out.println("1.-El barbero esta dormido por que es un webon");
33     System.out.println("2.-Entra cliente");
34     System.out.println("3.-Sale cliente");
35     System.out.println("4.-Esta lleno");
36     System.out.println("5.-Salir del sistema");
37     System.out.println("Elige opcion");
38
39 }
40 }
41 }
42 }
43 public class arreglos
44 {
45 {
46 int nummul;
47
48 static int lee_entero(String msj)
49 {
50 {
51
52 InputStreamReader stream = new InputStreamReader(System.in);
53 BufferedReader br = new BufferedReader(stream);
54
55 String cad;
56 int aux;
57
58 try
59 {
```



# UAEM

Universidad Autónoma  
del Estado de México



```
classess.java *arreglos.java numero.java ejercicio.java *menubarbero.java
59 {
60     System.out.print(msj);
61     cad = br.readLine();
62 } catch (java.io.IOException e)
63 {
64     cad = "0";
65 }
66 try
67 {
68     aux = Integer.parseInt(cad); //s3 convierte la cadena a entero
69 } catch (java.lang.NumberFormatException e)
70 {
71     aux=0;
72 }
73 return aux;
74 } //lo que hace resib una cadena y la convierte a entero
75
76
77
78
79
80
81
82
83
84
85
86
87
```

## Practica

Realizar un menu con operaciones, donde se selecciona una y tiene que hacer la operacion

```
menubarbero.java *operaciones.java arreglos.java
1 package miproyecto;
2 import java.io.*; //biblioteca que sirve para la entrada de datos y visualizacion a traves de la pantalla
3 public class operaciones { //inicia la clase principal
4
5
6
7
8     public static void main(String[] args) { //inicia el metodo principal
9         //declaracion de variable
10        variable var1=new variable(); //creacion de una nueva clase de tipo variable
11        //int opcion=0;
12        //int a=5,b=8;
13        //int suma,resta,multiplicacion,division;
14        //inicializacion y declaracion de clases
15
16        //cuerpo
17        System.out.println("OPERACIONES BASICAS");
18
19        System.out.println("menu");
20        System.out.println("1.-SUMA");
21        System.out.println("2.-RESTA");
22        System.out.println("3.-MULTIPLICACION");
23        System.out.println("4.-DIVISION");
24        System.out.println("Salir del sistema");
25        System.out.println("pulsa la opcion");
26        var1.opcion=lee_entero("\n");
27        switch (var1.opcion){
28        case 1:
29            System.out.println("1.-SUMA");
30            var1.a=lee_entero("dame el valor de a\n");
```



```
menubarbero.java *operaciones.java arreglos.java
30 System.out.println("el valor de a es :\n" + (var1.a));
31 var1.b=lee_entero("dame el valor de b es:\n" +(var1.b));
32 System.out.println("el valor de a es :\n" + (var1.a));
33 var1.resultado =var1.a+var1.b;
34 System.out.println("la suma es: " +(var1.resultado));
35 break;
36 case 2:
37 var1.a=lee_entero("dame el valor de a\n");
38 System.out.println("el valor de a es :\n" + (var1.a));
39 var1.b=lee_entero("dame el valor de b es:\n" +(var1.b));
40 System.out.println("el valor de a es :\n" + (var1.a));
41 var1.resultado =var1.a-var1.b;
42 System.out.println("la resta es: " +(var1.resultado));
43 break;
44 case 3:
45 var1.a=lee_entero("dame el valor de a\n");
46 System.out.println("el valor de a es :\n" + (var1.a));
47 var1.b=lee_entero("dame el valor de b es:\n" +(var1.b));
48 System.out.println("el valor de a es :\n" + (var1.a));
49 var1.resultado =var1.a*var1.b;
50 System.out.println("la multiplicacion es: " +(var1.resultado));
51 break;
52 case 4:
53 var1.a=lee_entero("dame el valor de a\n");
54 System.out.println("el valor de a es :\n" + (var1.a));
55 var1.b=lee_entero("dame el valor de b es:\n" +(var1.b));
56 System.out.println("el valor de a es :\n" + (var1.a));
57 var1.resultado =var1.a/var1.b;
58 System.out.println("la division es: " +(var1.resultado));
59 break;
60 }
61 }
62 }
63
64
65 static int lee_entero(String msj)//inicia la funcion lee entero
66 {
67
68     InputStreamReader stream = new InputStreamReader(System.in);
69     BufferedReader br = new BufferedReader(stream);
70
71     String cad;
72     int aux;
73
74     try
75     {
76         System.out.print(msj);
77         cad = br.readLine();
78     } catch (java.io.IOException e)
79     {
80         cad = "0";
81     }
82     try
83     {
84         aux = Integer.parseInt(cad);
85     } catch (java.lang.NumberFormatException e)
86     {
87         aux=0;
88     }
89 }
```



# UAEM

Universidad Autónoma  
del Estado de México



```
menubarbero.java *operaciones.java arreglos.java
69     BufferedReader br = new BufferedReader(System.in);
70
71     String cad;
72     int aux;
73
74     try
75     {
76         System.out.print(msj);
77         cad = br.readLine();
78     } catch (java.io.IOException e)
79     {
80         cad = "0";
81     }
82     try
83     {
84         aux = Integer.parseInt(cad);
85     } catch (java.lang.NumberFormatException e)
86     {
87         aux=0;
88         return aux;
89     }
90 }
91 }
92=class variable{
93     int a;
94     int b;
95     int resultado;
96     int opcion=0;
97 }
98 }
```

```
Console
<terminated> operaciones [Java Application] /usr/lib/jvm/java-1.7.0-openjdk-1.7.0.65.x86_64/jre/bin/java (6/5/2015 9:20:22)
1.-SUMA
dame el valor de a
6
el valor de a es :
6
dame el valor de b es:
05
el valor de a es :
6
la suma es: 11
```

## PRACTICA 2 INSTALACION DE XAMPP

**PROPOSITO:** el proyecto XAMPP permite instalar en Linux el servidor Web Apache, MySQL, PHP y Perl, esta diseñado específicamente para desarrolladores Web. Si lo instala en Red Hat o Fedora le permite crear un servidor web totalmente funcional para el sistema local o de red e iniciar el desarrollo de sus aplicaciones con facilidad.

### ALCANCES:

Configurar una red local

Iniciar, los servicios de XAMPP

Vizualizar una pagina web en la red local



## REQUERIMIENTOS

Sistema operativo centos v.6.5

XAMPP

Ambos de 32 o 64 bits (dependiendo la arquitectura)

## INSTRUCCIONES

Realiza la descarga de XAMPP y copiala en el escritorio, ua vez hecha esta actividad. Copia el archivo ejecutable en el directorio opt, ademas asigana los permisos necesarios para que el archivo pueda ser ejecutado con chmod, luego inicia la ejecucion.

Fig. 1 Se copia xampp en opt

```
rocio@localhost:/opt/lampp/htdocs/practicaxampp
Archivo Editar Ver Buscar Terminal Ayuda
eclipse-java-luna-SR2-linux-gtk-x86_64.tar.gz
jdk1.8.0_40
rh
xampp-linux-5.6.8-0-installer.run
xampp-linux-x64-5.6.8-0-installer.run
[rocio@localhost opt]$ chmod a+x xampp-linux-5.6.8-0-installer.run
[rocio@localhost opt]$ ./xampp-linux-5.6.8-0-installer.run
[rocio@localhost opt]$ ./xampp-linux-x64-5.6.8-0-installer.run
[rocio@localhost opt]$ su
Contraseña:
[root@localhost opt]# corazon
bash: corazon: no se encontró la orden
[root@localhost opt]# ./xampp-linux-5.5.24-0-installer.run
bash: ./xampp-linux-5.5.24-0-installer.run: No existe el fichero o el directorio
[root@localhost opt]# ls
eclipse
eclipse-java-luna-SR2-linux-gtk-x86_64.tar.gz
jdk1.8.0_40
rh
xampp-linux-5.6.8-0-installer.run
xampp-linux-x64-5.6.8-0-installer.run
[root@localhost opt]# chmod a+x xampp-linux-x64-5.6.8-0-installer.run
[root@localhost opt]# ./xampp-linux-x64-5.6.8-0-installer.run
[root@localhost opt]# ls
```



Fig.2 Ejecucion de XAMPP



Fig. 3 Instalación de XAMPP

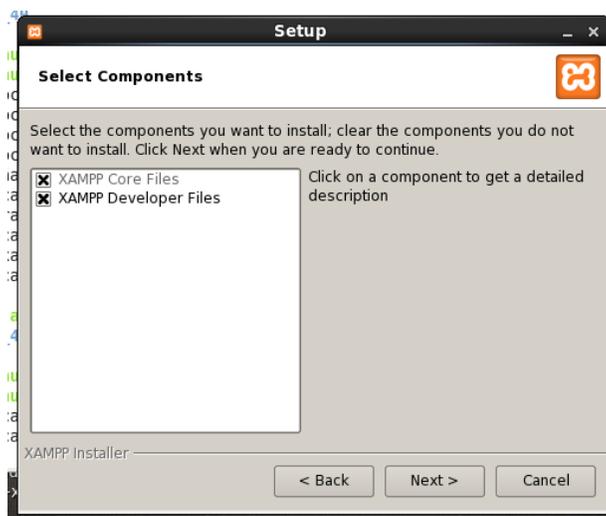


Fig.4 Instalacion se XAMPP



UAEM

Universidad Autónoma  
del Estado de México

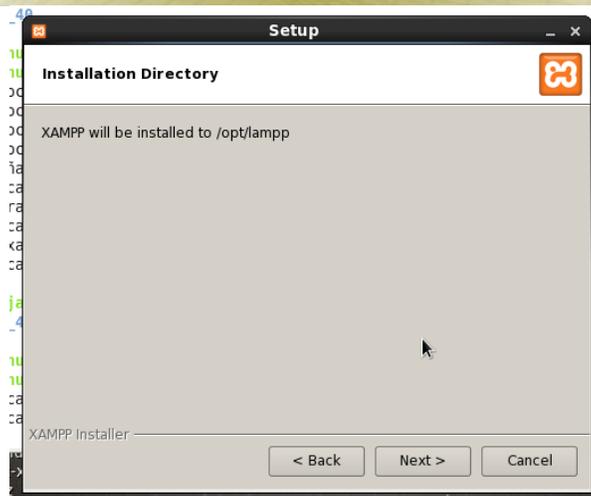


Fig. 5 Instalación de XAMPP





UAEM

Universidad Autónoma  
del Estado de México



Fig.6 Instalacion de XAMPP

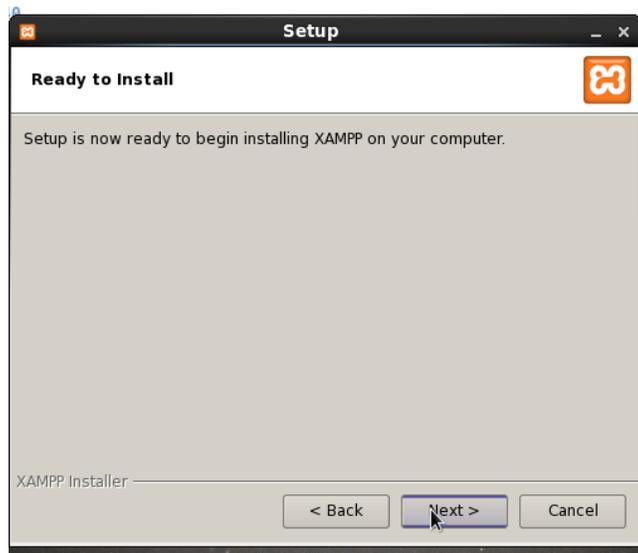


Fig.7 Instalacion de XAMPP



Fig.8 Instalacion de XAMPP



UAEM

Universidad Autónoma  
del Estado de México



Fig.9 Se finaliza la instalacion



Fig.10 Pantalla de inicio de XAMPP



UAEM

Universidad Autónoma  
del Estado de México



Fig.11 Activamos los servicios de XAMPP

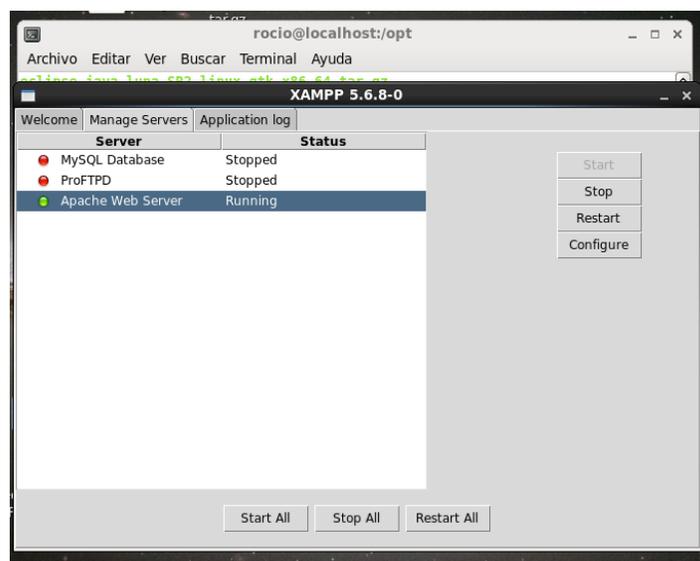


Fig.12 Entramos al localhost



UAEM

Universidad Autónoma  
del Estado de México



Fig.13 Entramos a la carpeta lampp, despues a htdocs donde creamos una carpeta llama practicaxampp y se creo un archivo llamado ejemplo.php.

```
MOZILLA FIREFOX
rocio@localhost:/opt/lampp/htdocs/practicaxampp
Archivo Editar Ver Buscar Terminal Ayuda
jdk1.8.0_40
lampp
rh
xampp-linux-5.6.8-0-installer.run
xampp-linux-x64-5.6.8-0-installer.run
[root@localhost opt]# cd la,pp
bash: cd: la,pp: No existe el fichero o el directorio
[root@localhost opt]# cd lampp
[root@localhost lampp]# ls
apache2  htdocs  licenses  php  uninstall
bin      icons   logs      phpmyadmin  uninstall.dat
build   img     man       proftpd     var
cgi-bin  include manager-linux-x64.run  properties.ini  xampp
ctlscript.sh  info  manual  RELEASENOTES
docs     lampp  modules  sbin
error    lib    mysql    share
etc      libexec pear

[root@localhost lampp]# cd htdocs
[root@localhost htdocs]# mkdir practica xampp
mkdir: no se puede crear el directorio «xampp»: El fichero ya existe
[root@localhost htdocs]# mkdir practicaxampp
[root@localhost htdocs]# cd practicaxampp
[root@localhost practicaxampp]# vi ejemplo.php
[root@localhost practicaxampp]#
```



UAEM

Universidad Autónoma  
del Estado de México



Fig.14 Escribimos un código donde se crea una página web que dice bienvenidos a mi página

```
rocio@localhost:~/opt/lampp/htdocs/practicaxampp
Archivo Editar Ver Buscar Terminal Ayuda
?php
echo "bienvenido a mi pagina web";
?>
```

"ejemplo.php" 3L, 44C

Fig.15 Dentro de localhost buscamos nuestra carpeta que se creó en htdocs y la seleccionamos



UAEM

Universidad Autónoma  
del Estado de México

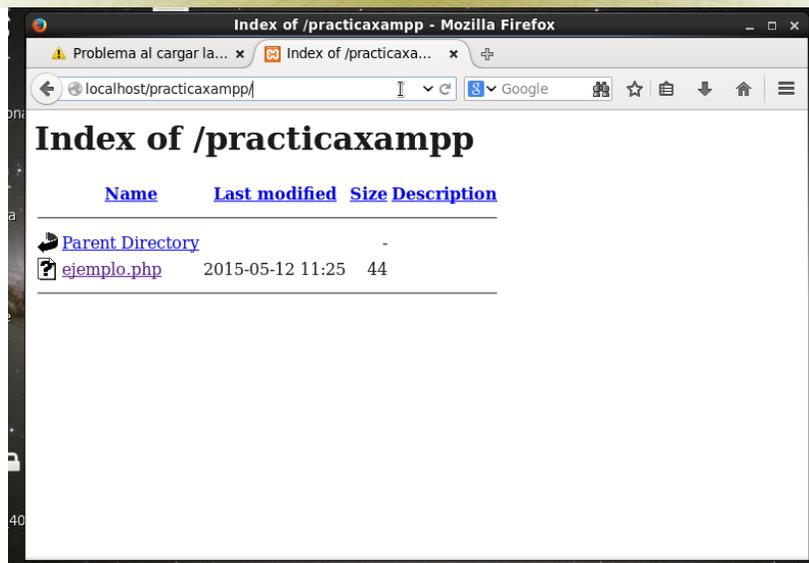


Fig.16 Se muestra nuestra primer pagina

