

UTN.BA

UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

Ingeniería en Sistemas de Información

Re Distinto

Documento de pruebas



Cátedra de Sistemas Operativos

Trabajo práctico Cuatrimestral

- 1C2018 -
Versión [1.2]

Requisitos y notas de la evaluación

Los requisitos expuestos a continuación se encuentran ampliados en [las Normas del Trabajo Práctico](#), que por practicidad, se han resumido a continuación.

Deploy y Setup

Es condición necesaria para la evaluación que **el Deploy & Setup del trabajo se realice en menos de 10 minutos**. Pasado este tiempo el grupo perderá el derecho a la evaluación.

Los archivos de configuración requeridos **para los diversos escenarios de pruebas** deberán ser preparados con anticipación por el grupo con todos los valores requeridos prefijados dejando los sólo los parámetros desconocidos (ej: IP) incompletos.

Los scripts ESI se podrán encontrar en su correspondiente [repositorio de github](#).

Compilación y ejecución

La compilación debe hacerse en la máquina virtual de la cátedra en su edición Server (no se pueden usar binarios subidos al repositorio).

Será responsabilidad del grupo verificar las dependencias requeridas para la compilación, y en caso de requerir bibliotecas provistas por la cátedra, descargarlas. También es responsabilidad de los integrantes del grupo conocer y manejar las herramientas de compilación desde la línea de comandos. Ver [Anexo - Comandos Útiles](#)

Evaluación

Cada grupo deberá llevar **una** copia impresa de la [planilla de evaluación](#)¹ **con los datos de los integrantes completos** (dejando los campos "Nota" y "Coloquio" en blanco) y una copia de los presentes tests.

Las pruebas **pueden ser alteradas o modificadas entre instancias de entrega** para verificar el correcto funcionamiento y desempeño del sistema desarrollado.

En estos casos el documento será actualizado y re-publicado para reflejar estos cambios.

¹ Al final de este documento

Pruebas

Prueba Mínima

Esta prueba representa la condición mínima para presentarse a la evaluación y sólo se evaluarán aquellos trabajos que finalicen esta prueba sin ningún inconveniente.

Configuración específica del sistema: Utilizar la siguiente configuración del sistema:

VM1: Planificador Algoritmo: SJF-SD Alfa: 50 Estimación inicial: 3 Claves bloqueadas: postre:ingrediente1	VM2: ESIs ESI_Largo ESI_MilanesaCompleta ESI_MultiClave ESI_Tiramisu ESI_MenuParrilla
VM3: Coordinador Algoritmo: EL Entradas: 50 Tamaño: 20 Retardo: 200	VM4: Instancias Algoritmo: CIRC Montaje: "/home/utnso/inst1/" Nombre: "Inst1" Intervalo de dump: 10 Algoritmo: CIRC Montaje: "/home/utnso/inst2/" Nombre: "Inst2" Intervalo de dump: 100 Algoritmo: CIRC Montaje: "/home/utnso/inst3/" Nombre: "Inst3" Intervalo de dump: 50

Actividades:

Una vez iniciado todo el sistema, detener la planificación para cargar los siguientes ESIs:

- a. ESI_Largo
- b. ESI_Tiramisu
- c. ESI_MultiClave

Una vez ingresados los 3 ESIs reanudar la planificación. Iniciar los siguientes ESIs:

- d. ESI_MilanesaCompleta
- e. ESI_MenuParrilla

Esperar a que arranque a ejecutar "ESI_MilanesaCompleta". Pausar la planificación. Desbloquear la clave "postre:ingrediente1" para todos los ESIs bloqueados. Reanudar y esperar la finalización

Resultados Esperados:

1. Los ESIs deberán finalizar sin inconvenientes en el siguiente orden:
 - a. ESI_Largo
 - b. ESI_MultiClave
 - c. ESI_MilanesaCompleta
 - d. ESI_MenuParrilla / ESI_Tiramisu
 - e. ESI_Tiramisu / ESI_MenuParrilla

Prueba Algoritmos de Reemplazo y Compactación

Esta prueba busca evaluar los diferentes algoritmos de reemplazo de la Instancia. Para esto se utilizará la siguiente configuración, pero se irán haciendo cambios en el algoritmo de la instancia (previo bajar el proceso instancia y volverlo a levantar).

<p>VM1: Planificador</p> <p>Algoritmo: SJF-SD Alfa: 50 Estimación inicial: 3 Claves bloqueadas: -</p>	<p>VM2: ESIs</p> <p>ESI_MultiReemplazo ESI_Compactador1</p>
<p>VM3: Coordinador</p> <p>Algoritmo: EL Entradas: 8 Tamaño: 10 Retardo: 300</p>	<p>VM4: Instancias</p> <p>Algoritmo: CIRC / LRU / BSU Montaje: "/home/utnso/inst1/" Nombre: "Inst1" Intervalo de dump: 10</p> <p>Algoritmo: CIRC / LRU / BSU Montaje: "/home/utnso/inst2/" Nombre: "Inst2" Intervalo de dump: 10</p>

Actividades:

Iniciar solamente la Inst1 y ejecutar el ESI_MultiReemplazo y verificar el orden de ejecución de los reemplazos.

Una vez finalizado el ESI_MultiReemplazo, bajar la Inst1 e iniciar la Inst2 y ejecutar el ESI ESI_Compactador2 y verificar en el instante que se inicien las compactaciones si es posible procesar otra operación (**Recomendación:** utilizar el comando Status <clave>).

Resultados Esperados:

Los resultados esperados para esta prueba varían según el algoritmo elegido, con lo cual deberán validarse con el ayudante al momento de correr la prueba. Los reemplazos deberán comenzar al terminar de ejecutar la sentencia `SET nintendo:consola:wiiu RIP`. Para poder chequearlo, es recomendable loguear los valores de la instancia ante cada sentencia, o bien pausar la planificación en este instante.

En algunos casos es posible que los resultados se tengan que analizar según como se hayan definido las resoluciones en caso de empate. Según los algoritmos, los resultados al ejecutar `SET sony:consola:ps3 TLOU` deberían ser:

	CIRC	LRU	BSU
0	FF8	FF8	Mario
1	GodOfWar	GodOfWar	FF8
2	TLOU	PkmnSnap	TLOU
3	FinalFanta	FinalFanta	FinalFanta
4	syTactics	syTactics	syTactics
5	MarioGalax	MarioGalax	MarioGalax
6	y2	y2	y2
7	RIP	TLOU	RIP

La compactación en este caso va a darse cuando se quieran insertar los últimos 3 valores ya que la distribución al momento de llegar a la instrucción `SET opcion3 pizzacuatroquesosgrande` es:

opcion1	opcion1	opcion1	opcion2	opcion2	opcion3	opcion3	opcion3
---------	---------	---------	---------	---------	---------	---------	---------

Y luego al llegar a la instrucción `"SET opcion3 pizzadepalmitos"` quedaría

opcion1	opcion1	Libre	opcion2	opcion2	opcion3	opcion3	Libre
---------	---------	-------	---------	---------	---------	---------	-------

Con lo cual para el SET subsiguiente se va a ver obligada la instancia a compactar.

Prueba Algoritmos de Distribución

Esta prueba busca evaluar los diferentes algoritmos de distribución del coordinador. Para esto se utilizará la siguiente configuración. Es importante tener en cuenta que se irán cambiando los algoritmos de distribución con lo cual habrá bajar todo el sistema y tendrán que cambiarse los archivos de configuración del Coordinador y volver a levantar todo.

VM1: Planificador Algoritmo: SJF-SD Alfa: 50 Estimación inicial: 3 Claves bloqueadas: -	VM2: ESIs ESI_Bar ESI_MilanesaCompleta ESI_MultiClave ESI_MultiReemplazo
VM3: Coordinador Algoritmo: EL / LSU / KE Entradas: 20 Tamaño: 10 Retardo: 250	VM4: Instancias Algoritmo: CIRC Montaje: "/home/utnso/inst1/" Nombre: "Inst1" Intervalo de dump: 10 Algoritmo: CIRC Montaje: "/home/utnso/inst2/" Nombre: "Inst2" Intervalo de dump: 100

Actividades:

Iniciar los ESIs en el orden detallado en el recuadro de la VM2 de ESIs y verificar en qué instancias se van asignando las claves y sus valores.

Resultados Esperados:

Los resultados esperados se irán validando según el algoritmo, pero debería verse en el caso de EL que las instancias se van llenando parejo (1 a 1) y en el caso de Key Explicit el ESI_Bar debería quedar todo junto en una sola instancia y el resto de las claves deberán ubicarse en ambas instancias y eventualmente habrá reemplazos.

Prueba Deadlock

Esta prueba busca evaluar el algoritmo de detección de deadlock

VM1: Planificador Algoritmo: SJF-SD Alfa: 50 Estimación inicial: 3 Claves bloqueadas: bebida13, segundoplato:ingrediente3	VM2: ESIs ESI_MenuCompleto ESI_Bar1 y ESI_Bar2 ESI_MilanesaCompleta ESI_MultiClave ESI_MultiReemplazo
VM3: Coordinador Algoritmo: EL Entradas: 50 Tamaño: 50 Retardo: 250 (Puede aumentar a 1250 para hacer más simple la prueba)	VM4: Instancias Algoritmo: CIRC Montaje: "/home/utnso/inst1/" Nombre: "Inst1" Intervalo de dump: 10 Algoritmo: CIRC Montaje: "/home/utnso/inst2/" Nombre: "Inst2" Intervalo de dump: 100

Actividades:

Iniciar el sistema, el ESI_MultiClave y el ESI_MulticlaveReemplazo.

Esperar que ejecuten algunas instrucciones e iniciar el ESI_MenuCompleto y ESI_Bar1.

Esperar hasta que estos se bloqueen e iniciar ESI_MilanesaCompleta y ESI_Bar2.

Una vez que no se ejecute nada por bloqueos desbloquear desde la consola las claves bebida13 y segundoplato:ingrediente3 y esperar que nuevamente quede todo el sistema bloqueado.

Consultar desde la consola del planificador si hay deadlocks.

Resultados Esperados:

Para el caso de los deadlocks deberían encontrarse los ESIs ESI_MenuCompleto, ESI_MilanesaCompleta, ESI_Bar1 y ESI_Bar2 en deadlock (ESI_MenuCompleto queda en deadlock con ESI_MilanesaCompleta y ESI_Bar1 queda en deadlock con ESI_Bar2, *no es necesario que se detalle cada uno de los deadlocks*).

Prueba Algoritmos de Planificación

Esta prueba busca evaluar los diferentes algoritmos de planificación del planificador. Para esto se utilizará la siguiente configuración. Es importante tener en cuenta que se irán cambiando los algoritmos de planificación con lo cual habrá bajar todo el sistema y tendrán que cambiarse los archivos de configuración del planificador y volver a ejecutar todos los procesos.

VM1: Planificador Algoritmo: SJF-SD / SJF-CD / HRRN Alfa: 50 Estimación inicial: 3 Claves bloqueadas: spoiler2, principal3, spoiler3, principal4	VM2: ESIs ESI_Spoilers ESI_Westworld ESI_Fargo
VM3: Coordinador Algoritmo: EL Entradas: 50 Tamaño: 50 Retardo: 250 (Puede aumentar a 1250 para hacer más simple la prueba)	VM4: Instancias Algoritmo: CIRC Montaje: "/home/utnso/inst1/" Nombre: "Inst1" Intervalo de dump: 10 Algoritmo: CIRC Montaje: "/home/utnso/inst2/" Nombre: "Inst2" Intervalo de dump: 100

Actividades:

Para SJF-SD

- Iniciar los ESIs en el orden especificado
- Esperar a que los tres se bloqueen
- Pausar, desbloquear "spoiler2", "principal3" y reanudar
- La estimación de ESI_Spoilers debería ser 2,5 y la de ESI_Westworld 4,5.
- Debería ejecutar ESI_Spoilers y bloquearse, ESI_Westworld y bloquearse, ESI_Fargo y finalizar.
- Pausar, desbloquear "spoiler3", "principal4" y reanudar
- La estimación de ESI_Spoilers debería ser 3,75 y la de ESI_Westworld 4,75.

Para SJF-CD

- Iniciar los ESIs en el orden especificado
- Esperar a que los tres se bloqueen
- Pausar, desbloquear "spoiler2", "principal3" y reanudar

- La estimación de ESI_Spoilers debería ser 2,5 y la de ESI_Westworld 4,5.
- Debería ejecutar ESI_Spoilers y bloquearse, ESI_Westworld y desbloquear "principal1". ESI_Fargo debería desalojarlo y bloquearse. Luego continúa ESI_Westworld y desbloquea "principal2". ESI_Westworld continúa hasta bloquearse. ESI_Fargo debería poder finalizar.
- Pausar, desbloquear "spoiler3", "principal4" y reanudar
- La estimación de ESI_Spoilers debería ser 3,75 y la de ESI_Westworld 4,75.

Para HRRN

- Quitar "principal3" de las claves inicialmente bloqueadas
- Iniciar los ESIs en el orden especificado
- ESI_Spoilers debería bloquearse.
- Mientras ejecuta ESI_Westworld, desbloquear "spoiler2" (lo más rápido posible). Esperar a que se bloquee.
- Deberían calcularse los RR de ESI_Spoilers y ESI_Fargo con la siguiente info:

ESI_Spoilers:	ESI_Fargo:
S = 2,5	S = 3
W = 0 a 10 (depende del desbloqueo)	W = 12
RR = 1 a 5	RR = 5

- Ejecuta ESI_Fargo y finaliza. ESI_Spoilers continúa hasta bloquearse.
- Pausar, desbloquear "spoiler3", "principal4" y reanudar
- ESI_Spoilers y ESI_Westworld deben finalizar en algún orden. El grupo deberá explicar qué orden y porqué.

Resultados Esperados:

Los eventos deberían ocurrir en el orden mencionado.

Prueba de Estrés

Esta prueba será la prueba final del trabajo práctico y buscará evaluar los límites del mismo para determinar si se tienen condiciones anómalas al funcionamiento

Configuración específica del sistema: Utilizar la siguiente configuración del sistema:

VM1: Planificador Algoritmo: SJF-SD Alfa: 50 Estimación inicial: 3 Claves bloqueadas: -	VM2: ESIs Todos los ESIs del repositorio podrán ser utilizados en esta prueba
VM3: Coordinador Algoritmo: EL Entradas: 50 Tamaño: 10 Retardo: 10	VM4: Instancias Algoritmo: CIRC Montaje: "/home/utnso/inst1/" Nombre: "Inst1" Intervalo de dump: 10 Algoritmo: LRU Montaje: "/home/utnso/inst2/" Nombre: "Inst2" Intervalo de dump: 10 Algoritmo: BSU Montaje: "/home/utnso/inst3/" Nombre: "Inst3" Intervalo de dump: 10

Dado que esta prueba busca explorar los límites del trabajo práctico para detectar los posibles errores no contemplados en las pruebas anteriores, no se tiene un flujo de actividad definido ni una serie de resultados esperados y *queda a criterio del ayudante a cargo de la evaluación* determinar el orden de ejecución de los ESIs y las acciones a tomar.

Anexo - Comandos Útiles

Copiar un directorio completo por red

```
scp -rpC [directorio] [ip]:[directorio]
```

Ejemplo:

```
scp -rpC tp-1c2015-repo 192.168.3.129:/home/utnso
```

Descargar bibliotecas en un repositorio (como las commons)

```
git clone [url_repo]
```

Ejemplo:

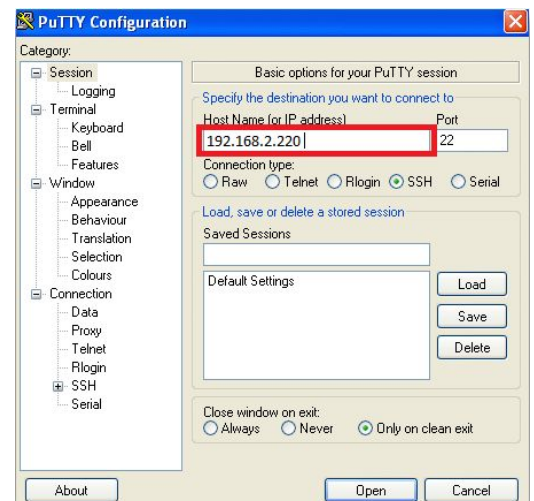
```
git clone https://github.com/sisoputnfrba/so-commons-library
```

PuTTY

Este famoso utilitario nos permite desde Windows acceder de manera simultánea a varias terminales de la Máquina Virtual, similar a abrir varias terminales en el entorno gráfico de Ubuntu.

Ya se encuentra en las computadoras del laboratorio y se puede descargar desde [aquí](#)

Al iniciar debemos ingresar la IP de nuestra máquina virtual en el campo **Host Name (or IP address)** y luego presionar el botón **Open** y loguearnos como **utnso**



Se recomienda investigar:

- Directorios y archivos: cd, ls, mv, rm, ln (creación de symlinks)
- Entorno: export, variable de entorno LD_LIBRARY_PATH
- Compilación: make, gcc, makefile
- Criptografía: md5sum
- Visor de procesos del sistema: htop

Sistema Completo

El deploy se hace de forma automatizada y en un tiempo límite de 10 a 15 minutos	
Los procesos ejecutan de forma simultánea y la cantidad de hilos y subprocessos en el sistema es la adecuada	
Los procesos establecen conexiones TCP/IP y se comunican mediante un protocolo	
El sistema no registra casos de Espera Activa ni Memory Leaks (Controlar mediante htop)	
El sistema responde de forma resiliente a la interacción con el entorno	
Se utilizaron de forma criteriosa los métodos estudiados para el manejo de múltiples conexiones (<i>multiplexado y arquitecturas multi-hilos</i>)	
El log permite determinar en todo momento el estado actual y anterior de los diversos procesos y del sistema junto con sus cambios significativos	
El sistema continúa su funcionamiento ante comandos erróneos o paths inexistentes (informándole al usuario el error)	
El sistema no requiere permisos de superuser (sudo/root) para ejecutar correctamente	

Proceso Planificador

Soporta múltiples ESIs conectados al simultáneo sin afectar la funcionalidad del sistema	
Se respeta el pasaje entre estados (colas) del sistema	
La unidad de estimación usada es una "sentencia"	
La ejecución bajo el algoritmo SJF Sin Desalojo se respeta	
La ejecución bajo el algoritmo SJF Con Desalojo se respeta	
La ejecución bajo el algoritmo HRRN se respeta	
La estimación de la próxima ráfaga se realiza mediante la media exponencial	
La liberación de claves respeta el algoritmo FIFO	
La consola permite pausar/continuar la planificación	
Es posible bloquear y liberar claves mediante la consola	
Es posible detectar correctamente un deadlock y finalizar un proceso interviniente	
La consola permite revisar el estado de cada clave	

Proceso ESI

Se ejecuta una única sentencia por cada mensaje del planificador	
--	--

El parser se encuentra integrado correctamente	
--	--

Interactúa correctamente con el Coordinador a la hora de ejecutar una sentencia	
---	--

Informa correctamente los errores detectados ante alguna situación de error prevista	
--	--

Ante la espera de ejecución por parte del Planificador, el proceso se mantiene bloqueado	
--	--

Proceso Coordinador

Las conexiones se gestionan mediante un servidor multihilos	
---	--

Soporta múltiples ESIs conectados al simultáneo sin afectar la funcionalidad del sistema	
--	--

Retorna con error al ESI si la clave consultada se encuentra bloqueada	
--	--

Se ejecuta el retardo de forma correcta antes de cada pedido de un ESI	
--	--

La operación GET no pasa más allá del Coordinador, creando la clave si no existe	
--	--

Permite la conexión y reconexión de múltiples Instancias	
--	--

Mantiene una tabla de instancias y la distribución de las claves en las mismas	
--	--

La distribución Equitative Load se respeta correctamente	
--	--

La distribución LSU se respeta correctamente	
--	--

La distribución Key Explicit se respeta correctamente	
---	--

La eliminación de una clave por baja de una instancia se realiza solamente ante un intento de acceder a la misma y solamente si la instancia no volvió a reconectarse	
---	--

El log de operaciones es claro y permite seguir la ejecución de las instrucciones	
---	--

Ante un pedido de compactación de una instancia el coordinador remite la orden a las demás instancias	
---	--

Proceso Instancia	
El tamaño es correctamente definido mediante una interacción con el Coordinador	
La tabla de entradas mantiene de forma consistente las claves-valor	
Se ejecuta a un intervalo regular la operación de dump de forma correcta	
Al reiniciar la instancia se toma en cuenta el dump para regenerar los valores de forma correcta	
Permite guardar tanto valores atómicos, como no atómicos	
Al no poseer más espacio, se ejecuta el algoritmo de reemplazo sobre los valores atómicos	
Se ejecuta un error al recibir un pedido sobre una clave reemplazada	
La ejecución del algoritmo de reemplazo Circular se ejecuta correctamente	
La ejecución del algoritmo de reemplazo LRU se ejecuta correctamente	
La ejecución del algoritmo de reemplazo BSU se ejecuta correctamente	
La compactación se ejecuta correctamente	
La instancia informa la necesidad de la compactación al coordinador	