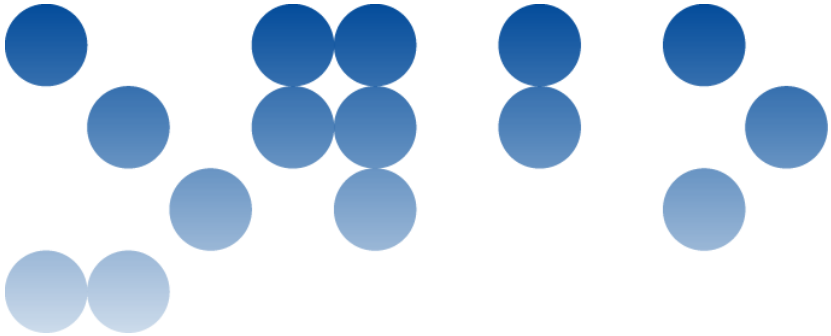

PRA1
Introducción a la programación en Linux

Sistemas Operativos

Programa

2016-01

Estudios de Informática, Multimedia y Telecomunicación

UOC Universitat Oberta de Catalunya

Presentación

Esta práctica plantea una serie de actividades con el objetivo que el estudiante pueda aplicar sobre un sistema Unix algunos de los conceptos introducidos en los primeros módulos de la asignatura.

El estudiante tendrá que realizar una serie de experimentos y responder las preguntas planteadas. También tendrá que escribir unos pequeños programas en lenguaje C.

La práctica se puede desarrollar sobre cualquier sistema Une (la UOC os facilita la distribución Ubuntu 14.04). Se aconseja que mientras realizáis los experimentos no haya otros usuarios trabajando al sistema porque el resultado de algunos experimentos puede depender de la carga del sistema.

Cada pregunta indica una posible temporización para poder acabar la práctica antes de la fecha tope y el peso de la pregunta a la evaluación final de la práctica.

El peso de esta práctica sobre la nota final de prácticas es del 40%.

Competencias

Transversales:

- Capacidad para adaptarse a las tecnologías y a los futuros entornos actualizando las competencias profesionales

Especificas:

- Capacidad para analizar un problema en el nivel de abstracción adecuado a cada situación y aplicar las habilidades y conocimientos adquiridos para abordarlo y resolverlo.
- Capacidad para diseñar y construir aplicaciones informáticas mediante técnicas de desarrollo, integración y reutilización.

Competencias propias:

- El uso y aplicación de las TIC en el ámbito académico y profesional.

Objetivos

Esta práctica tiene como objetivo que el alumno aplique los conceptos introducidos en los 4 primeros módulos de la asignatura sobre un sistema Unix y que se introduzca de forma progresiva en la programación, depuración y ejecución de aplicaciones en los sistemas Unix.

Restricciones en el uso de funciones de C

Para realizar la práctica debe utilizar las llamadas al sistema de Unix. **No se podrán utilizar funciones de C de entrada / salida** (getc, scanf, printf, ...), en su lugar se utilizarán las llamadas a sistema read y write. Sí que puede usar funciones de C para el formateo de cadenas (sprintf, sscanf, ...).

Enunciado

En esta práctica se deberá realizar una serie de experimentos y responder justificadamente a las preguntas planteadas. Puede hacer los experimentos sobre cualquier sistema Unix al que tenga acceso. Ahora bien, se aconseja que en ese momento no haya otros usuarios trabajando en el sistema.

Os facilitamos el fichero pr1so.zip con una serie de ficheros que os serán útiles para realizar la práctica. Descomprimirlo ejecutando el comando: unzip pr1so.zip. Para compilar un fichero, por ejemplo prog.c, hay que ejecutar el comando: gcc -o prog prog.c y para ejecutar el programa hay que hacer ./prog

1: Módulo 2 [Del 7 al 13 de Octubre] (5%+10%+10%=25%)

En este primer ejercicio vamos a analizar el comportamiento durante su ejecución de un programa en un sistema Unix. Para ello, os incluimos una serie de programas en c que deberá compilar y ejecutar siguiendo las instrucciones del ejercicio. En concreto, realizaremos las pruebas con un pequeño programa que implementa la multiplicación de dos matrices.

Para analizar el comportamiento de estos programas, utilizaremos el comando ***/usr/bin/time*** que muestra diversas estadísticas de utilización de recursos por parte de la aplicación y el tiempo requerido para su ejecución. En la figura 1, se muestra un ejemplo de la utilización de este comando y de las estadísticas que muestra. En este ejemplo la aplicación que ae monitoriza es el comando de ***sleep***, que únicamente realiza un retardo del número de segundos especificados como argumento. Tenga en cuenta que en el comando /***usr/bin/time*** se le pasa la opción -v para mostrar todas las estadísticas y que se ha de especificar el camino completo del comando.

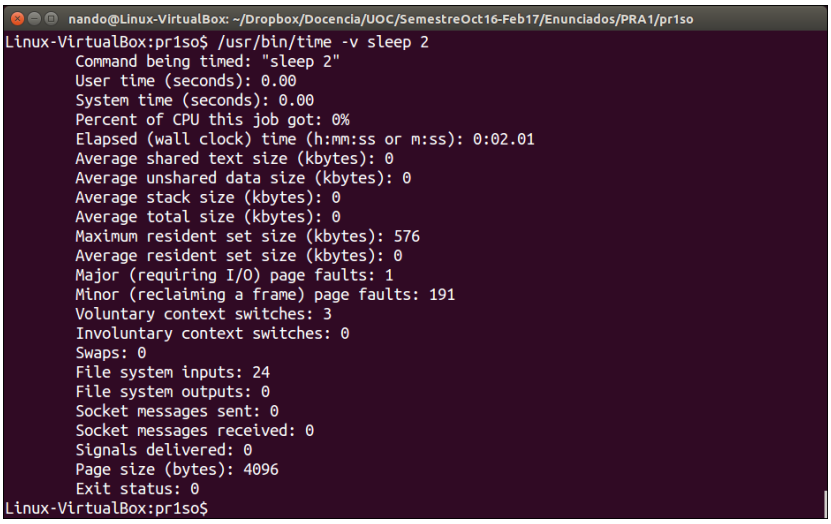

Figura 1. Ejemplo utilización comando / usr / bin / time

1.1. Ejecutar el comando "time" sin especificar el camino completo. Anexar la captura de pantalla (screen-shot) del resultado obtenido. Se obtiene el mismo resultado que el mostrado en la figura 1? En caso de respuesta negativa, indicar a que es debido la obtención de un resultado diferente. Consultad la sección "Shell Built Commands" de la ayuda del intérprete de comandos bash para responder a esta pregunta (man bash).

1.2. Os facilitamos el programa multmat.c, para calcular la multiplicación de dos matrices. Ejecutar este programa, junto con el comando ***/usr/bin/time -v*** (en el resto de subapartados también habrá que ejecutar los diferentes programas utilizando este comando). Anexar una captura de pantalla de las estadísticas de ejecución obtenidas. Interpretad el significado de las siguientes estadísticas, relacionándolas con los conceptos estudiados en la asignatura: "Command being timed", "User time", "System time", "Percent of CPU this job get", "Elapsed (wall clock) time ","Maximum residente set size", "Promedio residente set size", "Mayor (requiring I/O) page FAULTS", "Minor (Reclaiming a frame) page FAULTS", "Page size", y "Exit status ".

A partir de la información mostrada, responder a las siguientes preguntas:

a) ¿Qué sistema de gestión de memoria de los utilizados en teoría se está utilizando?

b) Justificar las diferencias existentes entre el user time y el walk clock time durante la ejecución?

c) Justificar a que son debidos los fallos de página obtenidos durante la ejecución del programa. Relacionarlo con el código del programa. ¿Por qué el número de fallos de página mayores es 0?

d) ¿Qué código de finalización devuelve el programa? ¿Qué significado tiene este código?

e) ¿Qué instrucción del programa habría que cambiar para modificar para que el código de retorno del programa fuese 22?

f) ¿Qué llamadas al sistema invoca el programa durante su ejecución?

g) ¿Cuál es el resultado mostrado por pantalla por el programa?

Observaciones:

- Para compilar el programa multmat.c, ejecutar el comando:

  > gcc multmat.c -o multmat

  y para ejecutar necesario utilizar las siguientes órdenes:

  > /usr/bin/time -v ./multmat
  > ./multmat

  en función de si queremos o no mostrar las estadísticas de su ejecución. El resto de programas de este apartado se compila y se ejecutan de forma análoga.

2: Módulo 3 [Del 14 al 26 de Octubre] (50%, los cinco apartados tienen el mismo peso)

2.1. Cambiar el tamaño de la matrices a multiplicar a 4000x4000, modificando la constante N. Recompilar y ejecutar de nuevo el programa multmat.c con el nuevo tamaño.

Anexar una captura de pantalla de las estadísticas de ejecución con el nuevo tamaño. Analizar las diferencias en las estadísticas de rendimiento.

¿A qué se debe el incremento de fallos de página menores? ¿Y los fallos de páginas mayores?

2.2. Utilizar ahora el programa multmat2.c, que es una variante del programa anterior (ambos con tamaño de matriz de 4000x4000). Adjuntar una captura de pantalla de las métricas de rendimiento obtenidas con /user/bin/time.

¿Comentar las diferencias existentes entre las dos ejecuciones?

Analizando el código de la nueva versión indicar a que se deben estas diferencias. Justificar la respuesta.

2.3. Modificar el programa original de multiplicación de matrices, para que las matrices se definan en la pila del proceso, en lugar de definirse como una variable global al programa. Incluir una copia del programa resultante, multmat3.c. Ejecutad la nueva versión y anexar una captura de pantalla de las estadísticas de su ejecución. ¿Se ejecuta correctamente la aplicación? ¿A qué se debe? Justificar la respuesta.

Para responder a esta pregunta puede ayudar el comando "**ulimit -a**" que muestra información sobre los límites de utilización de recursos definidos por el sistema para la ejecución de los procesos.

Utilizar el comando ***ulimit*** para permitir la ejecución de esta versión del programa multmat.c. ¿Qué parámetro se ha modificado y que valor le ha asignado? ¿Por qué? Anexar una captura de pantalla de la ejecución.

2.4. Modificar el programa original, para que ahora las matrices se almacenen en el heap (memoria dinámica). Incluir una copia del programa resultante, multmat4.c. Explicar los principales cambios introducidos a nivel de código para soportar la memoria dinámica. Ejecute la nueva versión y anexar una captura de pantalla de su ejecución. ¿En qué varía la nueva ejecución?

3: Módulo 4 [Del 27 de Octubre al 2 de Noviembre] (25%, los dos apartados tienen el mismo peso)

3.1. Utilizando exclusivamente llamadas al sistema de Linux, modificar el programa de multiplicación de matrices para que imprima el resultado (matriz R) por la pantalla. Incluir una copia del programa resultante (multmat5.c) y comentar el código añadido.

3.2. Con la nueva versión del programa (multmat5.c) y un tamaño de matriz de 1000x1000, ejecutar los siguientes comandos, analizando el tiempo de ejecución obtenido por cada uno de ellos:

- multmat5 > res.txt
- multmat5 > /dev/null
- multmat5 | awk '{s+=$1} END {print s}'

a) Indicar el tipo de dispositivos (lógicos, físicos, virtuales) utilizados en la ejecución de estos comandos.

b) Ordenar los comandos en función del tiempo de ejecución. Podéis utilizar el comando time del intérprete de comandos. Justificar el

porqué de dicha ordenación y las diferencias en los tiempos de ejecución entre los comandos.

Recursos

Recursos Básicos

- Documento "Introducción a la programación de Unix" (disponible en el campus virtual).
- Documento "Intérprete de comandos UNIX" (disponible en el aula) o cualquier otro manual similar.
- Cualquier manual básico de lenguaje C.
- El aula "Laboratorio de Sistemas Operativos" (puede plantear sus dudas relativos al entorno Unix, programación, ...).

Recursos Complementarios

- Manual de llamadas al sistema instalado en cualquier máquina UNIX (comando man).

Criterios de valoración

Cada uno de los apartados de la práctica tiene un peso del 50% sobre la puntuación final. El peso de cada apartado se distribuye de forma equitativa entre todos los subapartados que lo integran.

En la corrección se tendrán en cuenta los siguientes aspectos:

- Las respuestas deberán estar articuladas a partir de los conceptos estudiados en teoría y en la guías de la asignatura.
- Se valorará esencialmente la correcta justificación de las respuestas.
- Se agradecerá la claridad y la capacidad de síntesis en las respuestas.
- La capacidad de análisis de los resultados obtenidos y la metodología seguida para su obtención.
- El código entregado debe poder ejecutarse correctamente en cualquier máquina con Linux. Que el código se ejecute correctamente en su ordenador, no implica necesariamente que sea correcto. Revisar varias veces su solución y no ignorar los warnings que pueda estar generando el compilador.

Formato y fecha de entrega

Se creará un archivo zip que contenga un fichero pdf con la respuesta a todas las preguntas y los archivos .c con el código fuente de los distintos ejercicios.

El nombre del archivo tendrá el siguiente formato: "Apellido1Apellido2PRA1.zip". Los apellidos se escribirán sin acentos. Por ejemplo, el estudiante Marta Vallès y Marfany utilizará el nombre de archivo siguiente: VallesMarfanyPRA1.tar

La fecha límite para la entrega de la práctica es el **miércoles 2 de noviembre de 2016.**

Nota: Propietat intel·lectual

Sovint és inevitable, en produir una obra multimèdia, fer ús de recursos creats per terceres persones. És per tant comprensible fer-ho en el marc d'una pràctica dels estudis del Grau Multimèdia, sempre i això es documenti clarament i no suposi plagi en la pràctica.

Per tant, en presentar una pràctica que faci ús de recursos aliens, s'ha de presentar juntament amb ella un document en què es detallin tots ells, especificant el nom de cada recurs, el seu autor, el lloc on es va obtenir i el seu estatus legal: si l'obra està protegida pel copyright o s'acull a alguna altra llicència d'ús (Creative Commons, llicència GNU, GPL ...). L'estudiant haurà d'assegurar-se que la llicència que sigui no impedeix específicament seu ús en el marc de la pràctica. En cas de no trobar la informació corresponent haurà d'assumir que l'obra està protegida pel copyright.

Hauran, a més, adjuntar els fitxers originals quan les obres utilitzades siguin digitals, i el seu codi font si correspon.

Un altre punt a considerar és que qualsevol pràctica que faci ús de recursos protegits pel copyright no podrà en cap cas publicar-se en Mosaic, la revista del Graduat en Multimèdia a la UOC, a no ser que els propietaris dels drets intel·lectuals donin la seva autorització explícita.