

Fundamentos de Software - Prácticas

Módulo I. Órdenes UNIX y Shell Bash

Práctica 6: Depuración y control de trabajos

Departamento de Lenguajes y Sistemas
Informáticos – Universidad de Granada



UNIVERSIDAD
DE GRANADA

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:

- Conocer las principales características de depuración que ofrece bash y saber utilizarlas para detectar errores en un guion bash.

- Tipos de errores: léxicos, sintácticos, semánticos, efectos indeseados.

- Opciones de depuración:

1. Incluir la orden `echo` en puntos críticos

2. Opciones de depuración en bash:

- `-n`: Chequea errores sintácticos sin ejecutar el guion.
- `-v`: Visualiza cada orden antes de ejecutarla.
- `-x`: Actúa igual que `-v` pero de forma más abreviada.
- Posible uso al invocar el guión o dentro del mismo:

- `bash -x lee_archivo`

- `set -n ... set +n`

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:

- Conocer las principales características de depuración que ofrece bash y saber utilizarlas para detectar errores en un guion bash.

- Opciones de depuración:

- 3. Realizar la traza con la orden `trap`

- Especificar una acción a realizar como respuesta a un evento (recepción de una señal)
 - Invoca esta orden al comienzo del guion pasándole el argumento `DEBUG`: se ejecuta tras cada comando del guion.
 - Necesario invocarla en cada función.
 - `DEBUG`, `EXIT`, `ERR` y `RETURN` son ejemplos de lo que se denominan señales falsas. Ejecutan su código asociado en distintos momentos (tras cada comando, finaliza el guion, una orden devuelve un código de finalización distinto de cero, se regresa tras la ejecución de órdenes o guiones que se han ejecutado con `source`).

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:

- **Conocer las principales características de depuración que ofrece bash y saber utilizarlas para detectar errores en un guion bash.**
 - Opciones de depuración:
 - 4. Usar funciones de “aserción”.
 - Las funciones de aserción comprueban el valor de una variable en puntos críticos de la ejecución.
 - Lectura: <https://sites.google.com/site/jctovilla/programacion/c-c-para-linux-y-mac/aserciones-en-c-cassert-h>

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:
 - **Conocer los medios que ofrece la shell bash para controlar los trabajos de una sesión**
 - Trabajo/job: Órdenes que se mandan ejecutar al Shell.
 - Los trabajos pueden ejecutarse en distintos planos: primer plano (foreground), segundo plano background) o suspendido (detenido).
 - En cada sesión, la shell almacena una lista de los trabajos no finalizados (en segundo plano o suspendidos).
 - Ejecutar en segundo plano: incluir un & al final de la orden.
 - Podemos consultar: número de trabajo e identificador del proceso.
 - Los trabajos se pueden manipular usando órdenes de la shell.

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:
 - Conocer los medios que ofrece la shell bash para controlar los trabajos de una sesión
 - Los trabajos se pueden manipular usando órdenes de la shell.

Órdenes	Descripción
<code>jobs</code>	Lista los trabajos activos bajo el control del usuario (<code>help jobs</code>)
<code>fg</code>	Trae a primer plano un trabajo que se encuentra suspendido o en segundo plano (<code>help fg</code>)
<code>bg</code>	Envía a segundo plano un trabajo (<code>help bg</code>)
<code>%</code>	Permite cambiar el estado de un trabajo (<code>help %</code>)
<code>wait</code>	Espera la finalización de procesos en segundo plano (<code>help wait</code>)
<code>disown</code>	Suprime un trabajo de la lista de trabajos activos (<code>help disown</code>)
<code>kill</code>	Envía una señal a un/os proceso/s. Por defecto, finaliza la ejecución de un proceso (<code>man kill</code>)
<code>ps</code>	Muestra el estado de los procesos actuales en el sistema (<code>man ps</code>)
<code>top</code>	Muestra los procesos en ejecución con actualización de su información en tiempo real (<code>man top</code>)

Tabla 6.4. Órdenes de control de trabajos.

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:
 - Conocer los medios que ofrece la shell bash para controlar los trabajos de una sesión
 - Los trabajos se pueden manipular usando órdenes de la shell.

Especificador	Trabajo que es denotado con dicho especificador
%	Trabajo actual (%+ y %% son sinónimos de este especificador)
%-	Trabajo previo al actual
%n	Trabajo número n
%<cadena>	Trabajo cuya línea de órdenes comienza por <cadena>
%?<cadena>	Trabajo cuya línea de órdenes contiene <cadena>

Tabla 6.3. Especificadores de trabajos.

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:
 - Conocer los medios que ofrece la shell bash para controlar los trabajos de una sesión
 - Los trabajos se pueden manipular usando órdenes de la Shell:
 - `jobs`: ver el estado de los trabajos suspendidos y en segundo plano.
 - `fg`: trae a primer plano el trabajo especificado o el trabajo actual
 - `bg`: envía a segundo plano el trabajo especificado o el trabajo actual
 - `%`: abreviación de `fg` y `bg` (lleva a primer o segundo plano al trabajo actual o especificado según se acompañe de `&` o no).
 - `wait`: produce una espera hasta que el trabajo o proceso especificado como argumento haya finalizado su ejecución dicho proceso. Sin argumentos, la orden produce una espera hasta que todos los trabajos en segundo plano hayan finalizado.

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:
 - Conocer los medios que ofrece la shell bash para controlar los trabajos de una sesión
 - Los trabajos se pueden manipular usando órdenes de la Shell:
 - `disown`: elimina el trabajo cuyo identificador se da como argumento o el trabajo actual. No tiene efecto sobre trabajos que estén suspendidos (detenidos)
 - `kill`: también permite eliminar procesos, pero actúa tanto sobre procesos activos como suspendidos.
 - `kill -9`
 - `killall`

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:

- Saber utilizar la orden **ps** para conocer el estado de los procesos en ejecución.

- **ps**: muestra un listado de información detallada sobre los procesos actuales
 - Sin argumentos, muestra el listado de información sobre los procesos de la sesión actual.
 - Conocer estado de procesos concretos dando sus identificadores como argumento.

Opción	Efecto
-e	Muestra información de todos los procesos
-l	Muestra la información sobre los procesos en formato largo
-u <nombre>	Muestra el estado de los procesos del usuario <nombre>
-o <formato>	Permite usar un formato definido por el usuario para el listado

Tabla 6.5. Opciones de la orden **ps**.

Práctica 6: Depuración y control de trabajos

- OBJETIVOS:
 - **Saber utilizar la orden ps para conocer el estado de los procesos en ejecución.**
 - `top`: muestra una lista de procesos en ejecución de forma actualizada en tiempo real.
 - PID,
 - usuario al que pertenece,
 - consumo de CPU,
 - consumo de memoria RAM,
 - etc.

Práctica 6: Depuración y control de trabajos

- COMANDOS:

- jobs
- fg
- bg
- %
- disown
- kill/killall
- wait
- trap
- set
- sleep
- source
- top
- ps