

# LABORATORIO 1 de PROGRAMACIÓN 2

Curso 2010

## INTRODUCCIÓN A MODULA-2

### Índice

1	Introducción.....	2
2	Implementación.....	2
3	Intérprete de Comandos.....	2
3.1	Operaciones sobre el sistema.....	2
4	Módulos de definición.....	4
5	Librería Comando .....	4
6	Nueva información y comunicación.....	4
6.1	Uso del foro.....	5
7	Entrega.....	5
7.1	Plazos de entrega.....	6
7.2	Forma de entrega.....	6
7.2.1	Identificación de los archivos de las entregas.....	6
7.2.2	Re-entregas.....	7
7.2.3	Clave de acceso para entregar.....	7
7.2.4	Verificación de lo entregado.....	7
8	Individualidad.....	7
9	Compilador .....	8
10	Recursos del lenguaje.....	8
11	Evaluación.....	8

# 1 Introducción

Se desea implementar un sistema que reconoce comandos en la entrada estándar. El objetivo de este laboratorio es conocer el entorno de desarrollo utilizado en el curso. Los algoritmos a implementar, se resuelven aplicando los conceptos aprendidos en el curso de Programación 1.

## 2 Implementación

El programa principal tendrá como nombre *Interprete*.

## 3 Intérprete de Comandos

El programa **Interprete** lee comandos desde la entrada estándar y muestra la descripción de los mismos. La sintaxis general de los comandos es la siguiente:

*nombreComando parámetros ...*

Un comando está formado por un nombre seguido de una lista de parámetros eventualmente vacía. El nombre del comando está separado de los parámetros por **un (1)** espacio. Asimismo, cada parámetro se separa del siguiente por **un (1)** espacio.

Téngase en cuenta que para este laboratorio se puede asumir que **TODOS** los comandos se ingresan con una sintaxis correcta.

### 3.1 Operaciones sobre el sistema

El programa **Interprete** lee comandos y muestra su descripción hasta leer el comando *salir*.

Si el programa lee un comando que tiene parámetros, despliega en la salida estándar el siguiente mensaje:

Comando leído: [comando]

Con los siguientes parametros: [param1,param2,..., paramN]

Nota: la salida generada por el programa entregado debe ser **EXACTAMENTE IGUAL** a la indicada en la letra.

Ejemplo:

conectar terminal servidor

Comando leído: conectar

Con los siguientes parametros: terminal,servidor

Si el programa lee un comando que no tiene parámetros, despliega en la salida estándar el siguiente mensaje:

Comando leído: [comando]

Sin parametros

Ejemplo:

Comando leído: pausa

Sin parametros

Si el programa lee el comando *salir*, despliega en la salida estándar el mensaje Fin!, sin espacios ni saltos de línea luego del mensaje, y finaliza la ejecución del programa.

Ejemplo:

Fin!

Se presenta un ejemplo de la ejecución del programa:

sumar 4 3

Comando leído: sumar

Con los siguientes parametros: 4,3

imprimir datos

Comando leído: imprimir

Con los siguientes parametros: datos

conectar terminal servidor

Comando leído: conectar

Con los siguientes parametros: terminal,servidor

pausa

Comando leído: pausa

Sin parametros

sumar 1 2 3 4

Comando leído: sumar

Con los siguientes parametros: 1,2,3,4

salir

Fin!

## 4 Módulos de definición

El estudiante debe obtener los archivos correspondientes a los módulos de definición (*\*.def*) en un archivo *.zip* desde la sección Laboratorio del sitio web del curso:

*<http://www.fing.edu.uy/inco/cursos/prog2>*

**Los módulos de definición no se pueden modificar.** Tenga en cuenta que para el proceso de corrección se utilizarán los archivos *.def* publicados.

Las funciones, procedimientos y tipos de datos de los módulos de implementación deben coincidir con las definiciones correspondientes que se encuentran en los módulos de definición tal como aparecen descriptos en estos.

## 5 Librería Comando

El estudiante debe obtener la librería *Comando.lib* que procesa la entrada de los comandos y sus parámetros, en un archivo *zip* desde la sección Laboratorio del sitio web del curso. Las operaciones que provee esta librería se encuentran en el módulo de definición *Comando.def*, también publicado en el sitio web del curso.

Este módulo se entrega ya compilado, es decir, pronto para ser linkeditado con los restantes módulos del laboratorio.

## 6 Nueva información y comunicación

En caso de ser necesario se publicarán documentos con los agregados y/o correcciones al laboratorio que puedan surgir con el avance del curso.

Toda publicación de nueva información se realizará en la sección laboratorio del sitio *web* del curso, y será debidamente avisada en las clases teóricas, prácticas, foro y página *web* del curso.

Los estudiantes cuentan con las clases de consulta de laboratorio y el foro, para discutir acerca del laboratorio.

***Las casillas personales de los docentes no son el medio de comunicación adecuado para este tipo de discusiones, por lo cual se solicita no utilizarlo.***

## 6.1 Uso del foro

Antes de usar el foro del curso, es obligación leer el Reglamento de participación en los foros.

Cada mensaje debe ser enviado al foro que corresponda (dudas de letra, casos de test, programa principal, los distintos módulos, etc).

Para consultar dudas relacionadas a los casos de pruebas que se publiquen, el formato a utilizar para definir el asunto de un nuevo mensaje, es el siguiente:

- [nroCasodePrueba] Descripción breve

Antes de publicar un mensaje nuevo **verificar si ya no hay un hilo de conversación para el módulo, operación y descripción que se quiera consultar.**

Si lo hay entonces continuar sobre el hilo existente.

Sino iniciar un nuevo hilo según el formato definido anteriormente.

Al responder un mensaje, **NO INCLUIR** el texto que contiene el mensaje al que se responde, se puede leer el anterior siguiendo la conversación. Con esto se evita la acumulación y el tráfico de datos innecesarios, facilitando también la lectura de los mensajes.

***En caso de no respetar lo indicado anteriormente los docentes no contestarán dichos mensajes.***

## 7 Entrega

El estudiante deberá realizar la implementación del programa principal *Interprete.mod*.

El comportamiento de entrada y salida debe ser exactamente igual al descrito en la letra del laboratorio.

**No debe cambiarse la sintaxis de los comandos ni de las respuestas a los mismos.** Téngase en cuenta que esto será considerado en la evaluación, como un factor crítico para la aprobación del laboratorio.

***No se pueden realizar módulos adicionales a los solicitados.***

También se podrá entregar un archivo de texto con los comentarios que el estudiante crea necesarios. Este archivo deberá llamarse **leame.txt**.

El estudiante que no respete alguna de las indicaciones anteriores asume el riesgo de que su trabajo no sea (o no pueda ser) corregido y que en consecuencia se invalide la entrega, con la consiguiente pérdida del curso.

## 7.1 Plazos de entrega

El plazo para la entrega es hasta el lunes 22 de marzo del 2010, hasta las 8:00 horas.

Los trabajos deberán ser entregados dentro de los plazos indicados anteriormente. No se aceptarán entregas de trabajos fuera de fecha y hora. La no entrega o la entrega fuera de los plazos indicados, implica la pérdida del curso.

## 7.2 Forma de entrega

Las entregas se realizarán de forma automática vía *web* como ya se hizo en Programación 1. Para ello, se deberá acceder a la sección laboratorio del sitio *web* del curso y seguir los pasos que se explicarán en la página correspondiente.

Es importante señalar que para realizar las entregas se solicitará al estudiante la clave que utiliza para realizar los trámites de Bedelía.

Los estudiantes que se inscribieron en **forma individual** deberán ingresar la clave de bedelía para efectuar la entrega.

Los estudiantes que se inscribieron en **forma grupal**, para efectuar la entrega, deberán ingresar la clave de bedelía del integrante del grupo que se inscribió como representante, el otro estudiante no estará habilitado para realizar entregas.

Se recuerda que las computadoras de las salas 114 y 115 del 1<sup>er</sup> piso, así como las de las bandejas metálicas del Cuerpo Norte de Facultad tienen conexión a la red. Por lo cual, se puede acceder desde ellos a la página *web* para realizar las entregas.

*El estudiante que no respete alguna de las indicaciones que se detallan a continuación asume el riesgo que su trabajo no sea (o no pueda ser) corregido y que en consecuencia se invalide la entrega, con la consiguiente pérdida del curso.*

### 7.2.1 Identificación de los archivos de las entregas

Cada uno de los archivos a entregar debe contener en la primera línea del archivo un comentario con el número de cédula del estudiante **sin el guión y sin dígito de verificación**. Por ejemplo, si un estudiante tiene número de cédula 1910876-4 deberá escribir en la primera línea de cada uno de los archivos entregados el siguiente comentario:

(\* 1910876 \*)

IMPLEMENTATION MODULE nombreAr;

En el ejemplo anterior se asumió que se trata del archivo *nombreAr.mod*.

## 7.2.2 Re-entregas

Desde el día que se habilite el receptor de entrega del laboratorio y hasta la fecha de culminación de la entrega, el estudiante podrá realizar todas las re-entregas que considere necesario, teniendo en cuenta que sólo la última entrega será corregida.

Se recomienda realizar una primera entrega ‘de prueba’ de forma de verificar que puede realizar el procedimiento con normalidad. Recuerde que solo la última entrega será corregida.

## 7.2.3 Clave de acceso para entregar

Verifique que tiene o dispone de la clave que le entregó Bedelía antes de realizar la 1ª entrega. En caso de no tenerla comuníquese con Bedelía.

## 7.2.4 Verificación de lo entregado

La responsabilidad de verificar la integridad de cada entrega es del estudiante.

# 8 Individualidad

No existirán instancias en el curso para reclamos frente a la detección por parte de los docentes de trabajos de laboratorio no individuales, independientemente de las causas que pudiesen originar la no individualidad (a modo de ejemplo y sin ser exhaustivos: utilización de código realizado en cursos anteriores u otros cursos, perder el código, olvidarse del código en lugares accesibles a otros estudiantes, prestar el código o dejar que el mismo sea copiado por otros estudiantes, dejar la terminal con el usuario abierto al retirarse, enviarse código por mail, etc.), es decir, que se considera a cada estudiante **RESPONSABLE DE SU TRABAJO DE LABORATORIO Y DE QUE EL MISMO SEA INDIVIDUAL**.

Los trabajos de laboratorio que a juicio de los docentes no sean individuales **serán eliminados, con la consiguiente pérdida del curso**, para **todos** los involucrados. Además todos los casos serán enviados a los órganos competentes de la Facultad, lo cual puede acarrear sanciones de otro carácter y gravedad para los estudiantes involucrados.

Asimismo **se prohíbe el envío de código vinculado al laboratorio al foro del curso**, dado que el mismo será considerado como una forma de compartir código y será sancionado de forma severa.

## 9 Compilador

Todos los módulos entregados deben compilar y linkeditar con el compilador del curso (XDS-Modula2). De no ser así el trabajo será eliminado, con la consiguiente pérdida del curso.

El estándar a utilizar es el estándar ISO.

## 10 Recursos del lenguaje

Se podrán utilizar todas las herramientas del lenguaje que sean vistas en el curso, tanto en los teóricos como en los prácticos.

## 11 Evaluación

Se evaluará positivamente:

- Modularidad y estructuración del código.
- Cantidad y calidad de los comentarios.
- Claridad en el código.
- Uso de estructuras de datos adecuadas al problema.
- Nombre de las variables y constantes acordes a la función que desempeñan en el programa.

Se penalizará fuertemente:

- Uso de programación no estructurada.
- Uso de variables globales dentro de funciones o procedimientos.
- Programas que funcionen incorrectamente.

La tarea se evaluará con los ejemplos presentados en la letra del laboratorio, así como también con otros casos de prueba. En todos los casos el programa deberá funcionar correctamente de acuerdo a la especificación proporcionada.

Por último se recomienda que el estudiante tenga presente que: **la entrega de módulos y/o programas que, como mínimo, no compilen o linkediten, o cuya salida, en cuanto a su resultado, no sea exactamente idéntica a la de los ejemplos presentados en la letra del laboratorio, serán eliminados, con la consiguiente pérdida del curso.** Se recuerda que las tareas deben **compilar con el compilador del curso (Native XDS-x86 para Windows)** y **funcionar en las máquinas de los laboratorios de facultad (Windows XP).**