

Grado en Ingeniería Informática

Modelos de Computación

Práctica Número 1*

1 Ejecución de ILC

Durante gran parte del curso vamos a realizar las prácticas con el software **ILC**, desarrollado en la Universidad de Alicante. Se trata de un intérprete experimental del lenguaje **L** escrito en **C++**, y se presenta al usuario en versiones tanto para **Windows** como para **Linux**. La herramienta fundamental que incorpora el software es un intérprete del lenguaje teórico **L**, incluidas herramientas de

- Edición.
- Inclusión y expansión de macros.
- Interpretación del código de **L**.
- Simulación de Trazas de Computación.
- Codificación de programas mediante números de Gödel.

Para su ejecución es necesario tener localmente instalado el software, el cual puede ser descargado de la página web del proyecto **ILC** situada en

<http://ilc.sourceforge.net/>

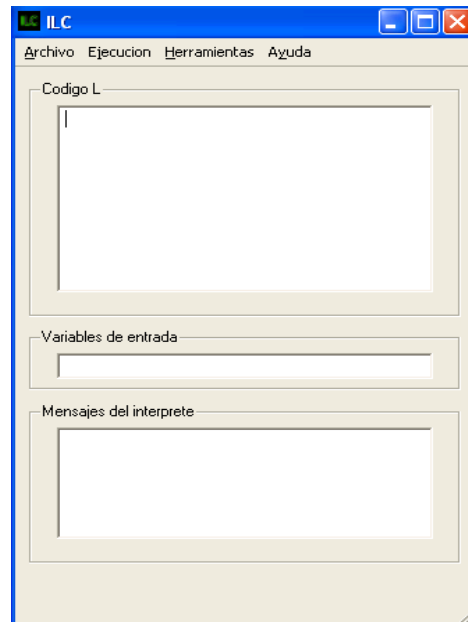
El siguiente texto tiene por objeto enseñarle los conceptos fundamentales de trabajo con **ILC**.

Para su ejecución es necesario hacer doble click sobre el icono



*© Antonio Tomen

o bien seguir la siguiente ruta, partiendo desde el botón de inicio: **Inicio**, **Todos los programas**, **ILC**, **ILC**. Se producirá la carga del programa, y obtendrá un entorno de trabajo como el que se muestra.



ILC en ejecución

Los elementos correspondientes al entorno de ejecución mostrado dividen al mismo en tres áreas principales. La primera, rotulada con la leyenda **Código L** es un sencillo editor que permite al usuario escribir directamente código L de acuerdo a las convenciones que su profesor le habrá explicado en clase de teoría¹, o también cargar un fichero de código L previamente salvado a disco. El segundo conforma una ventana lineal con la leyenda **Variables de Entrada**. En ella daremos valores iniciales a las variables de nuestro L-programa. La tercera, etiquetada **Mensajes del interprete** proporciona información procedente de la compilación indicando posibles errores u otras circunstancias.

Es muy conveniente que lea usted tanto la definición del lenguaje L que procesa el programa ILC, como un corto manual de instrucciones sobre el mismo. Ambos están disponibles en la página del proyecto. Navegue a la misma, y escoja los ítems de menú **Manual** y **Definición de L**, realizando a continuación una cuidadosa lectura de los mismos.

¹No obstante lo cual, existen ligeras variantes entre la sintaxis teórica de L, que usted ya conoce, y el correlato práctico de las mismas. Asegúrese de conocer tales diferencias, y de tenerlas en cuenta cuando edite código con ILC.

2 Edición de L-programas

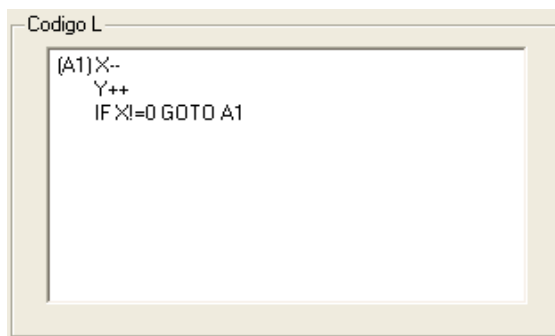
Vamos a realizar la escritura de nuestro primer programa. Aunque la sintaxis que utiliza ILC es muy parecida a la sintaxis del lenguaje teórico L, no es exactamente la misma. Quizás le convenga leer el corto manual de usuario que la página web de soporte le proporciona, donde podrá indicar especificaciones orientativas sobre las diferencias sintácticas, y que se encuentra disponible en el siguiente `url`:

`http://ilc.sourceforge.net/manual.html`

Es un programa que su profesor le ha presentado ya, y que sabemos que calcula la siguiente función:

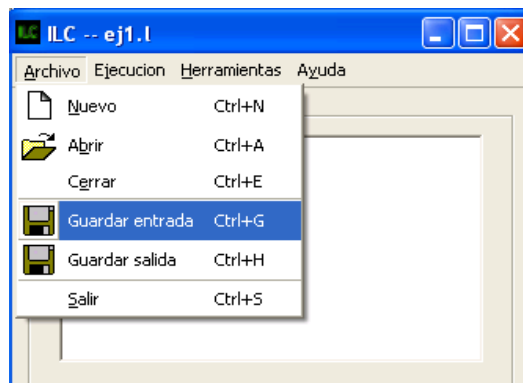
$$f(x) = \begin{cases} 1 & \text{si } x = 0 \\ x & \text{en otro caso} \end{cases}$$

Situése en la ventana de edición, y transcriba el programa tal y como se le propuso en clase de teoría, teniendo en cuenta las diferencias sintácticas citadas. Es un programa que hace uso de una etiqueta, que incorpora una bifurcación condicional, y cuyo aspecto final una vez transcrito en la ventana de edición se aprecia en la siguiente figura:



Editando un L-programa

El siguiente paso es guardar el fichero que hemos generado. Para ello escoja el ítem de menú **Archivo** y seleccione **Guardar entrada**. Se abrirá una ventana donde deberá indicar la carpeta donde desea guardar el fichero junto con el nombre que desea darle. Escoja como nombre `ident1.1` y guarde el fichero.

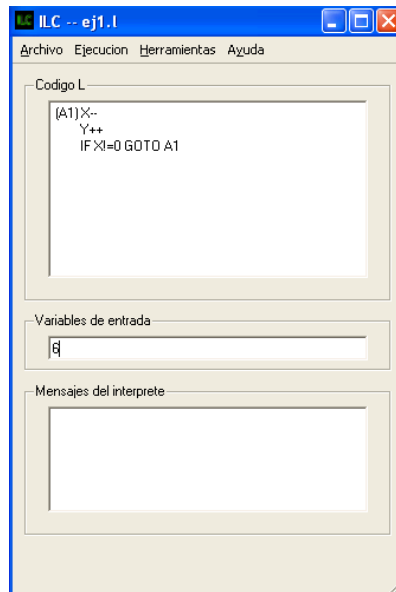


Guardando un L-programa

Por supuesto, también puede cargar programas editados previamente sin más que seleccionar el ítem **Abrir** del menú **Archivo**.

3 Ejecutando un L-programa

Una vez editado el L-programa, ha llegado el momento de dar valores a las variables de entrada, compilar el código y computar un resultado. Comenzamos por dar valores a la variable de entrada, utilizando la ventana adecuada para ello. Para cada variable de entrada, es necesario escribir en esta ventana el número natural que será su valor inicial. Para las variables de salida (Y) y locales (Z_i) esto no es necesario pues su valor inicial por convenio es cero. En el caso que nos ocupa, damos a la única variable de entrada el valor 6.



Dando valores a las variables

Y ya estamos en condiciones de realizar la computación sin más que escoger el ítem **Ejecutar** del menú **Ejecución** o pulsando directamente la tecla de función **F5**. Se realizará un proceso de análisis sintáctico a la búsqueda de errores y, si no los hay, se procesarán las variables de entrada con el programa editado para obtener el valor de la variable de salida. En nuestro caso, esperamos un valor de salida de 6. Compruebe que lo obtiene.

Si desea comprobar únicamente la corrección sintáctica del programa puede utilizar el ítem **Comprobar sintaxis** del menú **Herramientas**. Hágalo. Procedamos ahora a introducir algunos errores sintácticos a fin de que se familiarice con los posibles errores que puede cometer.

- modifique la primera instrucción por **(A1) X-**. Compruebe la sintaxis (Botón **F2**).
- modifique la segunda instrucción por **Y2++**. Compruebe la sintaxis.
- modifique la tercer instrucción por **IF X!=0 GOTO P**. Compruebe la sintaxis.
- establezca como valor de la variable de entrada **-1**. Compruebe la sintaxis.

La práctica le planteará otras situaciones parecidas que aprenderá a reconocer y solventar. En general, recuerde que para una función de n variables de entrada, deberá proporcionar esos valores en la ventana correspondiente separados por espacios en blanco.

4 Otras utilidades

ICL incorpora muchas más funcionalidades que usted irá aprendiendo en las siguientes prácticas. A continuación citamos algunas de ellas de carácter básico. Por ejemplo, podemos conocer el número de pasos (instrucciones básicas) ejecutados por un L-programa activando el flag **Contar Pasos de Ejecución** del ítem de menú **Ejecución**. Hágalo, establezca como valor inicial de la variable de entrada 5 y ejecute. Cambie ahora ese valor por 0 y vuelva a ejecutar. Es también posible activar el trazado de la ejecución del programa, codificar numéricamente programas o realizar la expansión de macros.

5 Ejercicios

(Único) En esta práctica se pretende familiarizarle con el entorno de L y con su uso para calcular funciones a través de instancias del modelo de cálculo. Se pide para los ejemplos de código L que se indican y que proporciona el propio *software* realizar las siguientes tareas:

- a) Editar el código e inspeccionarlo.
- b) Someterlo a una batería de datos de entrada que le permitan determinar los diferentes valores de salida. Escriba sus pruebas en una tabla.
- c) Utilizando la tabla anterior, determinar la función calculada por el código y especificarla en términos de dominio, codominio y expresión funcional.
- d) Si la función resulta ser parcialmente computable, explicar por qué lo es, y determinar al menos un punto del dominio donde la función no esté definida.

Genere un documento **.pdf** utilizando las "Normas de Edición" descritas en el bloque principal del curso que recoja para cada ejemplos de código toda la información generada en los apartados anteriores. Llámelo **Ejemplos_ILC.pdf**.

La lista de códigos de ejemplo sobre los que debe trabajar es accesible desde la siguiente ruta de ítems de menú de ILC: **Archivo->Abrir->Carpeta de Ejemplos**. Los ejemplos para los que debe desarrollar las tareas anteriores son: **ejx.1** siendo **x** cualquier número válido.

6 Rúbrica (Criterios de Calidad)

La rúbrica tiene por objetivo el que cuando usted entregue su trabajo, lo haga con unas mínimas garantías de corrección respecto al producto pedido. Para ello asegúrese, antes de subir al campus virtual sus desarrollos, de que éstos cumplen con las siguientes especificaciones:

- La función estimada para los programas de los ejemplos es correcta y la tabla de valores de entrada-salida es reproducible bajo la carga del código especificado.
- Se han respetado las convenciones de nombre, formato y contenido de todos los ficheros pedidos.

7 Procedimiento y Plazo de Entrega

La tarea de subida habilitada en *Moodle* le permitirá subir cada fichero que forma parte de los productos de la práctica, de forma individual y en el formato original. Para ello, suba el primer fichero de la forma habitual, y luego siga la secuencia de etapas que el propio *Moodle* le irá marcando. Recuerde además que:

- No debe hacer intentos de subida de borradores, versiones de prueba o esquemas de las soluciones. *Moodle* únicamente le permitirá la subida de los ficheros por **una sola vez**.
- La detección de plagio o copia en los ficheros de las prácticas, o la subida de ficheros vacíos de contenido o cuyo contenido no responda a lo pedido con una extensión mínima razonable, invalidará plenamente la asignación, sin perjuicio de otras acciones disciplinarias que pudieran corresponder.
- El plazo de entrega de la práctica se encuentra fijado en la tarea de subida del Campus Virtual.
- Entregas fuera de este plazo adicional no serán admitidas, salvo causa de fuerza mayor debidamente justificadas mediante documento escrito.
- Se recuerda que la entrega de todas las asignaciones de prácticas es recomendable, tanto un para un correcto seguimiento de la asignatura, como para la evaluación final de prácticas, donde puede ayudar a superar esta según lo establecido en la ficha de la asignatura.