

Antonio Pérez Carrasco

SRec versión 1.2: manual de uso

Número 2011-05

Serie de Informes Técnicos DLSI1-URJC

ISSN 1988-8074

Departamento de Lenguajes y Sistemas Informáticos I

Universidad Rey Juan Carlos

Índice

| | |
|--|-----------|
| 1. Visión general de la Aplicación | 3 |
| 1.1 Partes de la ventana | 4 |
| 2. Instalación de SRec | 5 |
| 3. Visualizaciones..... | 6 |
| 3.1 Vistas Disponibles | 6 |
| 3.1.1 Vista del Código | 6 |
| 3.1.2 Vista de la Traza de Ejecución | 6 |
| 3.1.3 Vista de la Pila de Control | 7 |
| 3.1.4 Vista del Árbol de Activación | 7 |
| 3.1.5 Vista Cronológica de la Estructura | 7 |
| 3.1.6 Vista de la Estructura de Datos..... | 8 |
| 3.2 Cargar y Procesar Clases | 8 |
| 3.3 Generar Visualizaciones | 9 |
| 3.3.1 Valores para parámetros | 10 |
| 3.3.2 Generación aleatoria de valores de parámetros..... | 11 |
| 3.4 Cargar Visualizaciones | 12 |
| 3.5 Guardar y Exportar Visualizaciones | 12 |
| 3.6 Cargar Animaciones GIF..... | 13 |
| 3.7 Guardar Traza en Formato HTML..... | 13 |
| 3.8 Manejar las Visualizaciones | 13 |
| 3.9 Opciones sobre la Visualización | 14 |
| 3.10 Opciones sobre los nodos | 15 |
| 4. Configuración de Visualizaciones | 16 |
| 4.1 Cantidad de Información que se desea Mostrar | 16 |
| 4.2 Clases de Datos que se desean Mostrar | 16 |
| 4.3 Ocultación de Datos Históricos y Subllamadas | 17 |
| 4.4 Visibilidad de la Estructura de Datos en Algoritmos “Divide y Vencerás”..... | 17 |
| 4.5 Formato de los Datos | 18 |
| 4.6 Zoom..... | 19 |
| 4.7 Ubicación de los Paneles y las Vistas | 20 |
| 5. Lectura y Edición de Código | 20 |
| 6. Otras opciones | 21 |
| 6.1 Selección de Máquina Virtual..... | 21 |
| 6.2 Generación de Archivos Intermedios..... | 21 |
| 6.3 Idioma | 22 |
| 6.4 Manejo de Configuraciones | 22 |
| 7. Disponibilidad de SRec | 23 |
| Agradecimientos | 23 |
| Referencias | 23 |

SRec versión 1.2: manual de uso

Antonio Pérez Carrasco¹

¹ Dpto. Lenguajes y Sistemas Informáticos I, Escuela Técnica Superior de Ingeniería Informática, Universidad Rey Juan Carlos, Móstoles, Madrid, España
antonio.perez.carrasco@urjc.es

Abstract. SRec es una herramienta destinada a la visualización animada de algoritmos recursivos y diseñados con la técnica de Divide y Vencerás con fines educativos. La aplicación ofrece al usuario múltiples vistas sobre la recursividad: traza, pila de control y árbol de recursión así como otras dos orientadas a Divide y Vencerás: vista cronológica y vista de la estructura de datos manejada por el algoritmo. SRec aporta algunas facilidades educativas, como por ejemplo un alto grado de flexibilidad en su configuración, el almacenamiento de visualizaciones generadas para su posterior recuperación o la exportación de las visualizaciones en formatos gráficos estándar. La versión 1.2 que presentamos integra todas las vistas implementadas hasta ahora, tanto las genéricas de la recursividad como las específicas de la técnica Divide y Vencerás, mejorando además algunos aspectos funcionales y de interfaz.

Keywords: visualización, animación, algoritmia, recursividad.

1 Visión General de la Aplicación

SRec permite visualizar de una manera gráfica, sencilla e intuitiva la ejecución de algoritmos recursivos mediante árboles de activación, aportando además otras vistas dinámicas tales como la pila de control y la traza de la ejecución, acompañadas de vistas adicionales sobre la estructura de datos sobre la que se trabaja en caso de algoritmos basados en la técnica de Divide y Vencerás. Además, en todo momento las vistas están acompañadas de la vista de código.

La aplicación permite configurar múltiples opciones sobre las visualizaciones destinadas a adaptarlas a los requerimientos específicos del usuario en cada momento. Además se permite almacenarlas en disco y volver a cargarlas sin necesidad de generarlas en cada ocasión, manteniendo todos los parámetros de configuración y el estado exacto en el que se encuentra la visualización en el momento de su guardado.

SRec también permite visualizar y modificar el código de un algoritmo, bien para corregir errores, bien para modificar el comportamiento de dicho código. Además se ofrece la posibilidad de generar animaciones en formatos gráficos estándar, aprovechables para páginas web y presentaciones electrónicas.

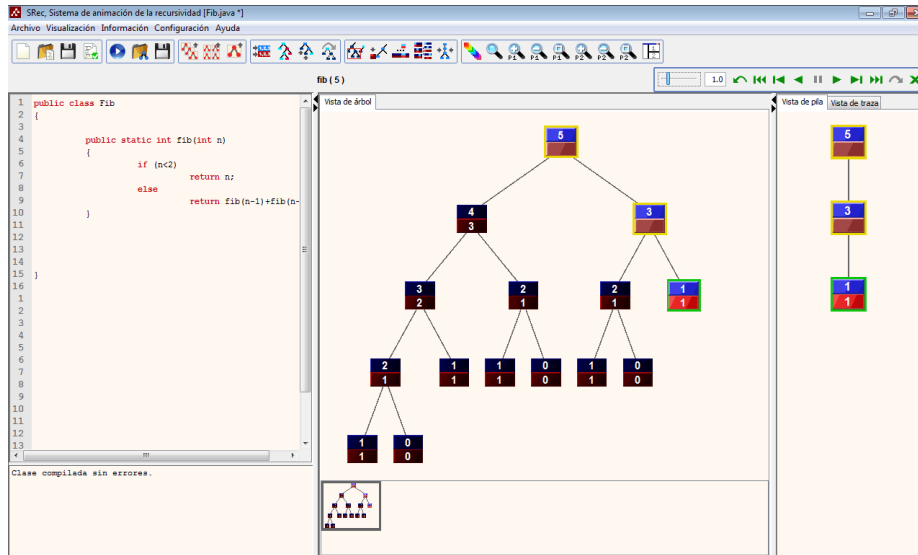


Ilustración 1 Ventana principal de SRec

1.1 Partes de la Ventana

La ventana de SRec está formada, como se puede ver en la Ilustración 1, por tres paneles, una barra de herramientas, una barra de animación y una barra de menús.

Los tres paneles sirven como medio de difusión de la información que SRec pretende transmitir a los usuarios. El panel situado en la parte izquierda ofrece el código de la clase Java cargada en la aplicación y a la que pertenece el algoritmo que se está visualizando. Este panel tiene un carácter estático en cuanto a su ubicación en la ventana, pues el único que no se puede reubicar en ninguna otra parte de la misma, si bien puede ser colapsado para aumentar el espacio dedicado a los dos paneles de visualización.

Los otros dos paneles ofrecen vistas dinámicas de las visualizaciones activas. Ambos pueden situarse uno encima de otro (tomando una forma horizontal) o bien uno al lado del otro, lo que potencia la verticalidad de los paneles, muy recomendable para vistas como la de la traza de la ejecución, que muestra el transcurso cronológico de subllamadas y resultados obtenidos hasta el momento en la ejecución del algoritmo.

Otra de esas vistas muestra la pila de control, que contiene aquellas llamadas recursivas que aún no han sido finalizadas y que, por lo tanto, se encuentran aún residentes en memoria. La representación de las subllamadas clona el formato de la vista principal, el árbol de activación. Ésta es la vista de mayor relevancia, ya que permite ver la ejecución completa del algoritmo a través del árbol para efectuar diferentes tipos de análisis. Cada nodo del árbol representa una subllamada recursiva y ofrece dos celdas: una, en la parte superior del nodo, para los valores de los parámetros de entrada para esa subllamada, y otra, en la parte inferior, para el valor de

retorno del método o para los parámetros de entrada con los valores actualizados tras la ejecución del algoritmo si se trata de un método que no devuelve un valor único.

SRec ofrece otras dos vistas más, específicas para la técnica de Divide y Vencerás. Esta técnica, cuando su representación está activa, acompaña opcionalmente cada nodo del árbol de activación con una representación destacada de la estructura de datos, de tal forma que se puede apreciar muy claramente qué partes de la misma se van tratando y cuáles son los resultados parciales obtenidos. Las dos vistas específicas se centran exclusivamente en dicha estructura; una ofrece una sucesión cronológica de los pasos de la visualización, en los que se puede ver cómo se va modificando el contenido de la estructura, mientras que la otra se centra en el estado actual de toda la estructura, aportando información sobre la jerarquía de subllamadas realizadas.

SRec cuenta con una barra de herramientas que permite acceder rápida y cómodamente a las funcionalidades de mayor interés del programa. Justo debajo está el área del título de la animación, que identifica la animación que se está visualizando gracias al nombre del método y los valores de los parámetros insertados por el usuario.

La barra de animación se encuentra justo a la derecha del área de título. Esta barra permite manejar la animación emulando los mandos de un reproductor de vídeo (avance, retroceso, parada...) y cerrarla.

2 Instalación de SRec

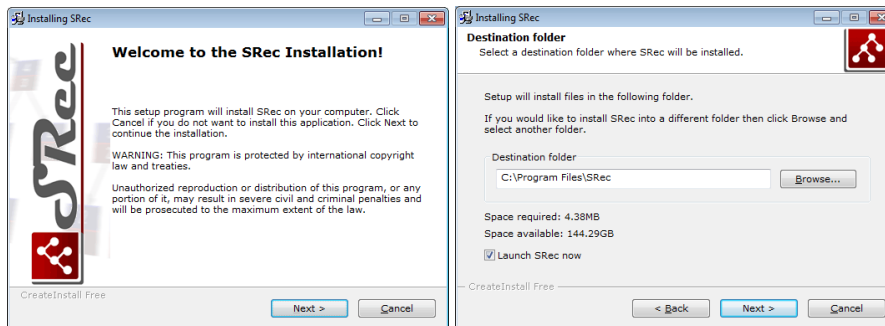


Ilustración 2 Pantallas del instalador

Conseguir que SRec funcione en nuestro equipo es muy sencillo, puesto que no necesita instalación. Los pasos son los siguientes:

1º Descargar la última versión disponible desde la web de SRec. Ésta vendrá dada en un archivo EXE (compatible únicamente por tanto con equipos Windows®).

2º Descomprimir en la carpeta deseada el contenido del archivo.

Para ejecutar SRec tan sólo es necesario abrir el enlace guardado en el menú de Inicio. Se necesita contar con la versión 6 del JDK (Java Development Kit) de Java.

3 Visualizaciones

SRec permite generar, cargar y almacenar las visualizaciones de los algoritmos recursivos que se deseen. En general, todas estas tareas resultan muy sencillas de realizar, puesto que no requieren más de 2 ó 3 pasos.

3.1 Vistas Disponibles

SRec ofrece un total de cinco vistas diferentes sobre cada visualización, en las cuales a medida que va progresando la animación van apareciendo datos nuevos.

La vista de código está disponible desde el momento en que esté cargada en la aplicación una clase para visualizar sus algoritmos. La vista de traza, pila de control y árbol de activación están disponibles en todas las visualizaciones que genera la aplicación. Las vistas cronológica y de estructura sólo se abren cuando estamos visualizando un algoritmo diseñado bajo la técnica de Divide y Vencerás, siempre que hayan sido habilitadas explícitamente por el usuario.

Todas estas vistas se detallan a continuación.

3.1.1 Vista del Código

La vista de código, la única vista estática de las que ofrece el programa, permite ver y editar el código de la clase que contiene el algoritmo que se está visualizando.

Se presenta escrito con múltiples colores, para diferenciar las palabras reservadas del lenguaje, los comentarios y el nombre del algoritmo (logrando así una fácil identificación del método en concreto y de las llamadas recursivas).

Incorpora un panel adherido en su parte inferior que permite conocer qué errores de compilación se han producido al intentar compilarla en caso de que la clase no sea correcta a nivel léxico, sintáctico o semántico. Señalando la línea donde aparece la línea del error el editor se situará automáticamente en dicha línea.

Este panel se puede configurar por parte del usuario en cuanto a colores, tamaño y tipo de fuente que se emplea.

3.1.2 Vista de la Traza de Ejecución

Gracias a esta vista, el usuario dispone en formato textual de un historial de las llamadas que se han realizado hasta el momento actual en que se encuentra la visualización. Esta información se va actualizando de manera sincronizada con el progreso de la animación.

Imita el formato y la configuración del árbol de activación, omitiendo llamadas finalizadas si se ha configurado SRec para que la información histórica no aparezca en la visualización o mostrándola atenuada si es así como se muestra en la vista del árbol.

En esta vista aparecen tanto los nombres de los métodos como los nombres de los parámetros en las líneas que representan las llamadas recursivas, para facilitar la comprensión de la ejecución del algoritmo recursivo.

3.1.3 Vista de la Pila de Control

Esta vista ofrece una visión independiente de la pila de control, que contiene todas las llamadas que se encuentran insertadas en ella pendientes de finalizar en cada momento. Estos nodos coinciden con los que aparecen remarcados en la vista del árbol de activación.

La vista de la pila mantiene en todo momento el mismo formato (colores, bordes estilizados de nodos...) que la vista del árbol de activación. Esto favorece la identificación y la comprensión de la información que se ofrece, en complemento a la vista del árbol.

3.1.4 Vista del Árbol de Activación

Ésta es la vista principal, donde el usuario puede ver de manera gráfica el desarrollo de toda la ejecución del algoritmo en forma de árbol de activación. Cada nodo representa una llamada recursiva del algoritmo, representando de manera diferenciada los valores de los parámetros de entrada (celda superior) y el valor de retorno (celda inferior).

Pulsando sobre el nodo, aparece una etiqueta flotante que informa de los valores de los parámetros y de retorno así como del método al que pertenece esa llamada. Pulsar sobre el nodo con el botón derecho del ratón abre un menú contextual que ofrece cuatro opciones: información detallada, información textual, posibilidad de marcar ese nodo como nodo activo y posibilidad de remarcar aquellos nodos que sean iguales al nodo seleccionado.

Algunos nodos aparecen remarcados con un marco de color. Esto quiere decir que se encuentran almacenados en la pila de control (son llamadas cuya ejecución no ha finalizado). Habrá un nodo que tendrá un color de marco distinto al resto. Será así como quede señalado el nodo activo, el nodo que está siendo tratado en el estado secuencial en que se encuentra la visualización en ese momento.

Esta vista contiene un visor de posición que permite situarse fácilmente sobre árboles extensos, con este visor podemos contextualizar rápidamente la porción del árbol que estamos viendo en detalle.

Los árboles que muestran llamadas a varios métodos diferentes ofrecen un acrónimo precediendo a los valores de los parámetros de entrada con el fin de distinguir las llamadas que se producen sobre uno y otro método. Pulsando sobre esos nodos se podrá ver en la etiqueta informativa el nombre completo del método así como los valores de los parámetros de entrada y de salida.

Según la configuración escogida, los árboles pueden aparecer enteros, con una parte atenuada (nodos históricos) o sólo con la rama actual visible. Además, pueden aparecer inicialmente comprimidos e ir aumentando su ancho a medida que va aumentando el número de subllamadas (nodos) visibles.

3.1.5 Vista Cronológica de la Estructura

Esta vista se centra exclusivamente en la visualización de la estructura de datos (array o matriz) que maneja el algoritmo, ignorando por tanto el resto de parámetros de los métodos. Paso a paso se van introduciendo en la vista representaciones de la estructura que permiten ver qué estados y qué valores va tomando la estructura.

En cada representación se puede ver la estructura completa, apareciendo marcada con colores la zona en la que está trabajando el algoritmo en la subllamada que se corresponde con esa representación.

Existen dos variantes. Una da prioridad al tiempo, haciendo que la sucesión de representaciones que muestran o bien subllamadas nuevas o bien alcance de resultados se vayan mostrando en estricto orden cronológico. La otra mantiene ligada la salida de cada subllamada a su entrada; las subllamadas se van representando en orden cronológico según se van efectuando, pero los valores de salida siempre aparecen justo debajo o al lado (según el tipo de estructura) de los valores de entrada.

3.1.6 Vista de la Estructura de Datos

Al igual que la vista cronológica, la vista de estructura omite todos los parámetros de las diferentes subllamadas excepto la estructura de datos (array o matriz) que maneja el algoritmo. Sin embargo, a diferencia de la vista anterior, esta vista sólo muestra el estado actual de la estructura; dicho de otro modo, no muestra una sucesión de representaciones sino una representación única y con más información añadida.

Así, la vista se ayuda de colores para identificar las partes que ya se han calculado y las partes que aún no se han sometido a los efectos del algoritmo. Además, mediante líneas, se pueden ver las jerarquías de llamadas que se realizan, bien sobre la propia representación de la estructura, bien en la parte inferior mediante un sencillo e intuitivo diagrama.

3.2 Cargar y Procesar Clases

Para poder generar visualizaciones nuevas, es necesario procesar la clase Java en la que se encuentra el algoritmo que deseamos visualizar. Este proceso, que tan sólo toma unos pocos segundos, permite a la aplicación generar un código Java alternativo que incluye las sentencias necesarias para la creación de una estructura de datos interna que contendrá la traza de la ejecución. Esta traza tiene como objetivo almacenar todas las llamadas recursivas que se realicen durante la ejecución del algoritmo seleccionado, junto con los valores de los parámetros de entrada y de su valor de retorno.

Para iniciar el proceso, es necesario seleccionar la opción *Archivo / Cargar y Procesar clase...* A lo largo de este proceso, SRec pregunta si se desea activar las vistas específicas de *Divide y Vencerás* para algún método concreto. Para cada método habrá que indicar, tras señalarlo, qué parámetro contiene la estructura que se quiere visualizar y qué parámetros delimitan las posiciones de la estructura sobre la que actuará cada subllamada del algoritmo. SRec ofrece un cuadro de diálogo muy gráfico para facilitar la inserción de estos datos, es mostrado en la Ilustración 3.

Tras ello, la aplicación genera una nueva clase que es compilada, quedando lista para múltiples ejecuciones, que tendrán lugar cada vez que el usuario genere una nueva animación.

No estarán disponibles para su procesado y ejecución aquellos métodos que no tengan ningún parámetro de entrada así como los que tengan como parámetro o valor de retorno estructuras de más de dos dimensiones o tipos no primitivos. Tampoco podrán ejecutarse métodos que estén declarados dentro de una clase abstracta.

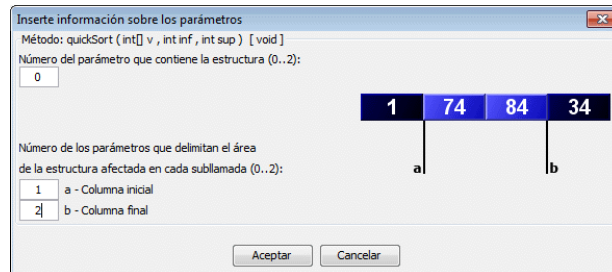


Ilustración 3 Diálogo para procesar adecuadamente los métodos con algoritmos de la técnica “divide y vencerás”

La aplicación lanzará un mensaje advirtiendo la existencia de errores (incapacidad de escribir en disco, errores de compilación de la clase de entrada...) si se encontraran durante el proceso. En caso contrario, a partir de entonces, podrán ser generadas tantas visualizaciones como se deseen sobre los métodos de la clase procesada.

3.3 Generar Visualizaciones

Para crear una nueva visualización, debe dirigirse a la opción *Archivo / Nueva animación* una vez esté debidamente procesada la clase a la que pertenece el método que se desea visualizar. SRec mostrará un cuadro de diálogo que permitirá escoger un método entre aquellos que fueron procesados (cuadro grande de fondo de los dos de la Ilustración 4).

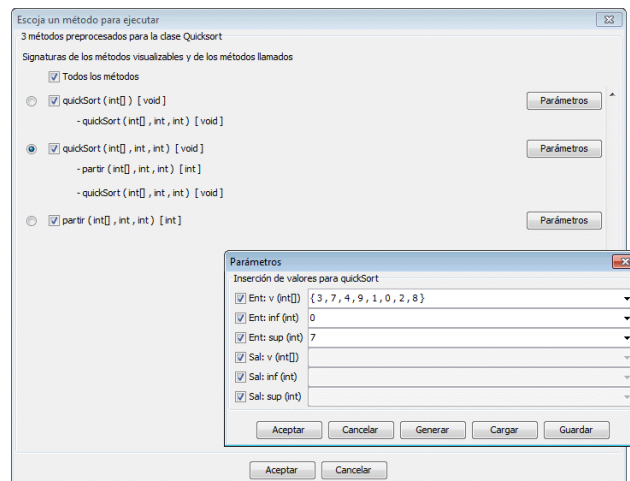


Ilustración 4 Diálogos de selección de método e inserción de parámetros

Para ejecutar el método, debe seleccionarlo con el botón de radio para que la aplicación sepa que ése es el método cuya ejecución debe lanzar. La casilla de verificación de cada método debe estar marcada si queremos que el método sea

visible en la animación en caso de que sea llamado desde el método que inició la animación o desde cualquier otro que forme parte de la misma.

SRec requiere que se introduzca por parte del usuario el conjunto de valores para los parámetros del método escogido. Para ello, se debe pulsar el correspondiente botón *Parámetros*. El cuadro que aparece (el pequeño traído a primera línea en la Ilustración 4) contendrá tantos cuadros de texto habilitados como parámetros de entrada tenga el método. En cada uno deberá escribirse un valor apropiado en función del tipo al que pertenezca el parámetro. La manera de escribir correctamente los valores se comenta en el subapartado siguiente “Valores para parámetros”.

Es posible que no interese mostrar el valor de alguno de los parámetros durante la visualización. Para ello, el cuadro permite ocultar parámetros no deseados mediante las cajas de verificación situadas a la izquierda, por lo que consecuentemente no aparecerá en la visualización. Al menos un parámetro debe permanecer siempre seleccionado para poder realizar la animación.

El cuadro de inserción de valores permite además cargar y guardar valores. Para cargar valores almacenados en el sistema de archivos previamente, tan sólo es necesario pulsar el botón *Cargar* y seleccionar el archivo adecuado. Si el archivo contiene valores para otro método o no es un archivo válido, la aplicación lo advertirá debidamente. Para guardar los valores actualmente escritos en los campos de texto se debe pulsar el botón *Guardar* y ubicar el nuevo fichero en disco.

La aplicación da la facilidad de generar aleatoriamente el valor de uno, varios o todos los parámetros de un método. Se detalla esta funcionalidad, a la que se accede tras pulsar el botón *Generar*, en el subapartado “Generación aleatoria de valores de parámetros”. Por si fuera poco, el cuadro de diálogo también ofrece la posibilidad de seleccionar un valor introducido con anterioridad durante la sesión actual; basta emplear el cuadro de texto desplegable para seleccionar el valor ya usado que se quiere volver a emplear.

Una vez que se han introducido correctamente los valores para los parámetros y se pulsa *Aceptar*, la aplicación ejecuta dicho método con los valores proporcionados, mostrando su resultado tras breves instantes. En este punto, la visualización ya está generada, se encuentra situada en el estadio inicial (llamada principal abierta) y se puede utilizar y configurar con total flexibilidad.

3.3.1 Valores para los Parámetros

Antes de realizar la ejecución del método, es necesario que el usuario introduzca un valor válido por cada uno de los parámetros de los que consta dicho método.

La manera de introducir valores válidos para cada uno de los tipos de parámetros que admite la aplicación es la siguiente:

Byte: cualquier número entero que entre en el rango (-128..127).

Short: cualquier número entero que entre en el rango (-32768..32767).

Int: cualquier número entero que entre en el rango (-2147483648..2147483647).

Long: cualquier número entero que se encuentre dentro de los límites del rango definido como (-9223372036854775808..9223372036854775807).

Float: número con parte decimal opcional (separada de la parte entera por un punto '.'). No se admite notación científica.

Double: número con parte decimal opcional (separada de la parte entera por un punto '.'). No se admite notación científica.

Char: un único carácter alfanumérico. Se admiten además eñes y vocales acentuadas con el acento utilizado en la lengua española.

String: se permite la introducción de cadenas de caracteres alfanuméricos, eñes y vocales acentuadas con el acento utilizado en la lengua española.

Boolean: se debe introducir la cadena "true" o "false" (sin comillas). Se admite cualquier combinación de mayúsculas y minúsculas.

Para la inserción de arrays, es necesario encerrar los valores entre llaves y separar cada elemento de los demás mediante comas (','). Se permite una cantidad variable de espacios en blanco entre elementos, comas y llaves, pero no de manera interna en un elemento.

Ejemplos válidos de arrays:

Array de enteros (int): { 400, 74, 28, -823 }

Array de coma flotante (float): { 4.7, 7, 2.777778, -8.7 }

Para la inserción de matrices, se debe introducir un array de arrays. Esto es, se debe encerrar una sucesión de arrays (escritos con las mismas reglas expuestas anteriormente) entre llaves, separando cada uno de los arrays por comas.

Ejemplos válidos de matrices:

Matriz de enteros (int): { { 400, -434, 46 }, { 4100, 1228, -991 } }

Matriz de coma flotante (float): { { 14.1, 177.11 }, { 2.777778, -8.7 }, { 5.666, 6.555 } }

3.3.2 Generación Aleatoria de Valores de los Parámetros

Para poder generar de manera aleatoria los valores de los parámetros cuando se está generando una visualización, se debe pulsar el botón *Generar* del cuadro de inserción de valores de parámetros. Una vez dentro del cuadro de diálogo, mostrado en la Ilustración 5, el usuario puede seleccionar todos los parámetros o únicamente algunos de ellos: sólo se generarán valores aleatorios para los parámetros que estén seleccionados en el momento de la pulsación del botón *Aceptar*.

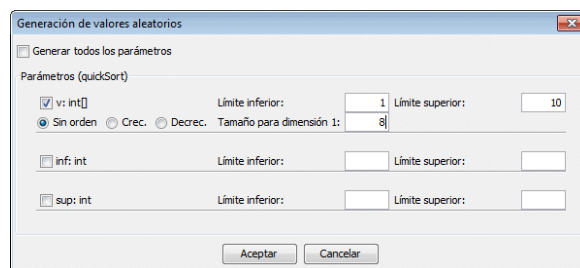


Ilustración 5 Cuadro de generación aleatoria de valores de parámetros

El cuadro permite establecer umbrales inferiores y superiores para cada parámetro. Aunque la inserción de esos valores es opcional está altamente recomendado introducirlos. Si no se insertan valores, se tomarán los límites que permiten los propios tipos de parámetros.

No se permite la introducción de límites superiores que tengan un valor menor al límite inferior introducido. No se permite tampoco la introducción de valores que

pertenezcan a otros tipos (por ejemplo, no está permitido delimitar la generación de enteros mediante la introducción de un valor de coma flotante).

Para arrays y matrices se permite -no es obligatoria- la introducción de valores para el tamaño. Si no se introduce ningún valor en el cuadro correspondiente, los arrays y matrices tomarán tamaños aleatorios (se recomienda indicar el tamaño). No está permitida la introducción de valores de coma flotante o negativos para la determinación del tamaño de arrays o matrices.

3.4 Cargar Visualizaciones

La aplicación permite cargar una visualización guardada previamente por la propia aplicación. Para cargarla y poder visualizarla, tan sólo es necesario seleccionar la opción *Archivo / Cargar animación*.

Tras seleccionar el archivo en el que se encuentra almacenada la visualización, la aplicación lee todos los datos sobre la visualización, las opciones de configuración y de formato, por lo que se recupera así toda la información útil para mostrar la visualización exactamente igual que aparecía en el momento del guardado.

Una vez que la animación está cargada, ésta se puede utilizar y configurar con total flexibilidad.

3.5 Guardar y Exportar Visualizaciones

Si al usuario le interesa guardar una visualización, con un estado concreto y una configuración de opciones y formato determinados, puede hacerlo seleccionando la opción *Archivo / Guardar animación*. Una vez hecho esto, se guarda en un archivo toda la información de interés que permite volver a mostrar la visualización en el estado exacto en el que se guardó, incluyendo las opciones de configuración.

Para exportar una animación en formato GIF tan sólo hay que seleccionar la opción *Archivo / Exportar animación como GIF*. La aplicación reiniciará la visualización y la recorrerá hasta el final de manera visible. Cuando termine, generará el archivo determinado por el usuario, el cual contendrá la animación generada en formato GIF. Se recomienda no hacer un uso intensivo de la CPU del ordenador mientras se realiza el salvado de la animación.

La aplicación da la posibilidad de guardar en disco una captura de cualquiera de las vistas disponibles en la ventana en tres formatos diferentes: JPEG, GIF y PNG (éste último ofrece el mejor resultado de los tres formatos). Para guardar en disco un archivo gráfico con la captura del árbol actualmente representado basta seleccionar la opción de menú *Archivo / Exportar captura*. Tras seleccionar la ubicación del fichero y el formato, SRec generará la captura en apenas un segundo.

SRec es capaz además de guardar en cualquiera de los tres formatos citados una captura por cada estado de la animación gracias a la opción *Archivo / Exportar capturas de animación*. Es decir, tras seleccionar esta opción SRec comienza a desarrollar la animación desde el principio, pero en vez de guardar un único archivo GIF animado, almacenará en disco tantos archivos gráficos como pasos tenga la animación. Cada archivo muestra un estado concreto de la animación.

Los archivos gráficos de los formatos JPG, GIF y PNG pueden ser incrustados en documentos de ofimática y páginas web, entre otros tipos de archivos electrónicos. Las animaciones GIF por su parte pueden ser insertadas en páginas web ya que son soportadas por todos los navegadores.

3.6 Cargar Animaciones GIF

SRec es capaz de cargar las animaciones GIF que él mismo ha creado con anterioridad. Para ello basta seleccionar la opción *Archivo / Cargar animación GIF* y seleccionar el archivo gráfico que contiene la animación.

La animación se iniciará automáticamente y no permitirá ningún tipo de interactividad. Una vez que llegue a su fin, la animación permanecerá parada hasta que se decida cerrar la animación.

Los paneles de la vista de código, la vista de traza y la vista de pila permanecerán vacíos en todo momento y las opciones de configuración del programa no tendrán efecto alguno en la animación.

3.7 Guardar Traza en Formato HTML

La aplicación permite guardar en un archivo web HTML el contenido actual del panel de la traza. Esta funcionalidad se lanza al seleccionar la opción *Archivo / Guardar traza* y determinar la ubicación del archivo HTML que se generará.

La apariencia de la página web creada será muy similar a la del panel. Además, contendrá el nombre del método y los parámetros iniciales para indicar a qué ejecución corresponde esa traza.

3.8 Manejar las Visualizaciones

Una vez que la visualización está preparada, se habilita un panel en la aplicación que muestra el nombre del algoritmo (método) junto con los parámetros que se han utilizado en la ejecución.



Ilustración 6 Barra de animación

La visualización permite navegar gracias a la barra de animación que se muestra en la Ilustración 6 a través de ella hacia delante (séptimo botón) y hacia atrás (tercer botón), mostrando paso a paso la evolución de la ejecución. Además, se puede reinicializar la visualización colocándola al comienzo de la misma (segundo botón), o bien adelantarla hasta el final (octavo botón), cuando el resultado final de la ejecución es visible.

También se pueden activar animaciones automáticas de tal forma que la visualización, cada un determinado periodo de tiempo (normalmente menos de cinco

segundos) dé un paso hacia adelante o hacia atrás, según se desee (acceso mediante los botones sexto y cuarto, respectivamente). Las animaciones pueden ser paradas en cualquier momento (quinto botón).

Ese periodo se puede ajustar mediante la barra deslizante o escribiendo el número de segundos en el cuadro de texto (se admiten valores decimales). Entre las acciones posibles, también está la que permite llevar a cabo un salto en la visualización de la animación que permite adelantar (noveno botón) o retroceder (primer botón) de una sola vez todas las subllamadas contenidas en el nodo activo en ese momento. Una vez que queramos cerrar la visualización, tan solo es necesario pulsar el botón de cierre correspondiente al panel de la visualización (décimo botón).

Los botones se muestran de color verde cuando su pulsación tiene algún efecto sobre la animación. Toman color rojo cuando su pulsación no tiene sentido o no tiene efecto alguno sobre la visualización.

Cuando el ratón entra en uno de los botones, éste toma el color naranja, color que también es tomado por los botones cuarto y sexto cuando las animaciones están activas.

3.9 Opciones sobre la Visualización

SRec proporciona algunas funcionalidades referidas a las visualizaciones que se detallan a continuación.

Por un lado, SRec permite realizar búsquedas de llamadas mediante el cuadro de diálogo que aparece en la Ilustración 7. Aquellos nodos que cumplan los criterios de la búsqueda se mostrarán resaltados en un color diferente. Los criterios de búsqueda se basan en el método al que pertenece el nodo y los parámetros de entrada y/o salida que se insertan para el citado método.

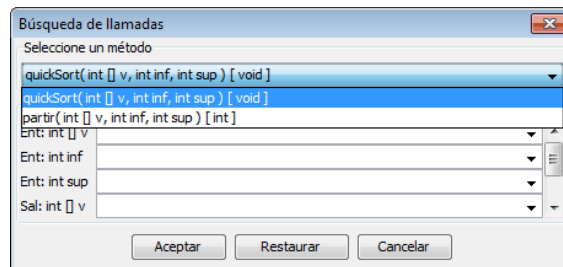


Ilustración 7 Diálogo de inserción de criterios de búsqueda de llamadas

El usuario tiene la posibilidad de dejar en blanco uno, varios o todos los campos de texto donde insertar los valores de búsqueda. De esta forma, se entiende que no se quiere discriminar por el valor de esos parámetros, por lo que aparecerán resaltados todos los nodos que cumplan sólo con el método escogido y los parámetros para los que sí se haya insertado un valor de búsqueda.

Por otra parte, la aplicación ofrece un cuadro resumen de la visualización actualmente abierta, mostrado en la Ilustración 8, donde se pueden ver ciertos valores

numéricos referentes al número de nodos totales, número de nodos visibles, número de nodos ocultos, número de nodos totalmente ejecutados...

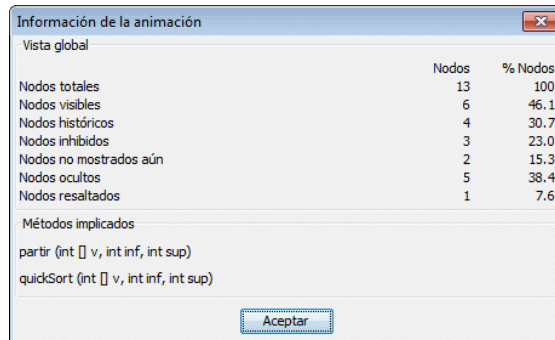


Ilustración 8 Diálogo de información sobre la visualización

El cuadro de información también ofrece el listado de métodos involucrados en la animación, mostrando la interfaz completa de cada uno de ellos.

3.10 Opciones sobre los Nodos

Los nodos de la vista del árbol de activación son sensibles a su pulsación con los botones izquierdo y derecho del ratón.

El botón izquierdo provoca la aparición de una etiqueta informativa que contiene el nombre del método al que pertenece la subllamada que representa el nodo. Esa etiqueta contiene también los valores conocidos de los parámetros de entrada y de salida.

El botón derecho despliega un menú con 4 opciones:

- *Detalle E/S*: muestra una etiqueta similar a la que aparece al pulsar el nodo con el botón izquierdo, pero ésta es más completa ya que contiene el nombre de cada parámetro y además ofrece aquellos parámetros que el usuario ha decidido que no resulten visibles en la animación.
- *Hacer nodo activo*: Esta opción persigue que el nodo pinchado se convierta en el nodo activo de la animación, por lo que ésta retrocede hasta alcanzar el estado en que se inicia la subllamada representada por ese nodo.
- *Explicación textual*: se llama a un cuadro de diálogo que ofrece numerosos datos sobre el nodo en cuestión, como el número de hijos que tiene, el número de subnodos que cuelgan de él, el estado en que se encuentra, etc.
- *Buscar llamadas iguales / Restaurar*: Ambas opciones se alternan en función de si el nodo se encuentra resaltado o no. La primera opción permite resaltar todos los nodos que pertenezcan al mismo método que el nodo apuntado por el ratón y que además contengan los mismos valores en sus parámetros de entrada. La segunda opción permite restaurar los nodos para que tengan la apariencia habitual y dejen de estar resaltados.

4 Configuración de las Visualizaciones

Existen múltiples opciones que SRec permite configurar según las preferencias o necesidades del usuario.

4.1 Cantidad de Información que se desea mostrar

Al crear una nueva animación la aplicación ofrece la posibilidad de ocultar métodos en las animaciones, pero ésta es una funcionalidad que permanece disponible una vez que la animación ya está creada.

Mediante la opción *Visualización / Métodos y parámetros*, SRec abre un cuadro de diálogo, representado en la Ilustración 9, que permite variar los nodos que se visualizan así como los valores de entrada y salida que se muestran.

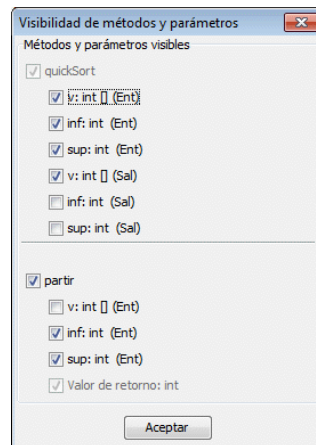


Ilustración 9 Diálogo de visibilidad de métodos y parámetros

De esta manera, podremos filtrar subllamadas de determinados métodos y hacer que sólo se muestren los valores de aquellos parámetros en los que estemos interesados.

En todo momento se debe garantizar que para cada método visible debe haber al menos un parámetro de entrada así como el valor de retorno del mismo. Si el método no retorna ningún valor, la aplicación da la opción de mostrar los valores de esos parámetros tras la ejecución del algoritmo, si bien se puede escoger no ver el valor de ningún parámetro en la salida. La aplicación no permite eliminar de la animación el método principal, es decir, el método que lanzó la ejecución.

4.2 Clases de Datos que se desean Mostrar

La aplicación permite elegir qué clase de datos deben aparecer en la visualización. Así, pueden mostrarse los valores de los parámetros de entrada junto con los valores

de salida, sólo los valores de los parámetros de entrada o bien únicamente los valores de salida.

Para determinar esta configuración es necesario seleccionar la opción *Visualización / Datos de Entrada y Salida...* y seleccionar los valores que se desea que aparezcan en la visualización. Los cambios tendrán efecto de manera inmediata sobre la visualización. Bastará pulsar *Aceptar* para salir del cuadro.

4.3 Visibilidad de Datos Históricos y Subllamadas saltadas

La aplicación da la posibilidad mediante el cuadro de diálogo visible en la Ilustración 10 de elegir entre mantener datos en la visualización utilizados en pasos anteriores o irlos atenuando o eliminando según van dejando de ser trascendentes en el estado actual de la visualización.

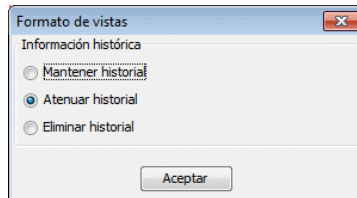


Ilustración 10 Diálogo para seleccionar el tratamiento de los nodos históricos

El formato en que se muestran los valores atenuados es configurable por el usuario. Para determinar esta configuración es necesario seleccionar la opción *Visualización / Nodos históricos...* y seleccionar cualquiera de las tres posibilidades que plantea el cuadro de diálogo.

Por otro lado, la aplicación permite escoger entre mostrar el subárbol de las subllamadas que se generan cuando se da un salto sobre la llamada recursiva actual, o bien mostrar sólo el resultado. Este subárbol se mostrará siempre y cuando la configuración actual no elimine de la visualización los nodos históricos. Basta seleccionar o deseleccionar la opción *Visualización / Mostrar subárboles en saltos...* para que los efectos se hagan notorios.

4.4 Visibilidad de la Estructura de Datos en Algoritmos “Divide y Vencerás”

El usuario puede elegir entre ver o no ver la estructura de manera destacada en cada nodo del árbol que represente un método de la técnica Divide y Vencerás. Sólo tiene que seleccionar o deseleccionar la opción *Visualización / Mostrar estructura en la vista del árbol* para que los efectos tengan lugar.

La vista cronológica de la estructura también es configurable desde el menú Visualización. Así, a través de la opción *Visualización / Mostrar estructura completa en la vista cronológica* da la opción de que aquellas partes que no están siendo manejadas por el algoritmo en cada subllamada aparezcan atenuadas o desaparezcan.

Por otro lado, se puede escoger el orden en que aparecen los resultados en la vista de estructura: estos pueden aparecer en estricto orden cronológico, intercalados con las distintas activaciones de subllamadas que van teniendo lugar, o bien aparecer ligadas a la entrada de la subllamada a la que pertenecen. Seleccionar uno de los dos modos de visualización es posible gracias a la opción *Visualización / Mostrar la salida ligada a la entrada en la vista cronológica*.

4.5 Formato de los Datos

Existe total flexibilidad a la hora de dar formato a los datos que se muestran en la visualización. Por un lado, a través de la opción *Visualización / Formato tipográfico...* se pueden aplicar distintos colores y formatos a las celdas de los valores de los parámetros de entrada y a las de los valores de salida.

La primera pestaña está dedicada a la configuración de los colores del modo 1, que permite diferenciar las celdas de cada nodo por colores. Las celdas que contienen valores de entrada tendrán un color y las que contengan valores de salida, otro. La segunda pestaña, visible en la Ilustración 11, recoge el repertorio de colores para el modo 2, que permite diferenciar a los nodos completos unos de otros en función del método al que representan. Ambos modos de coloreado permiten visualizar las celdas con degradados para hacer la visualización más agradable.

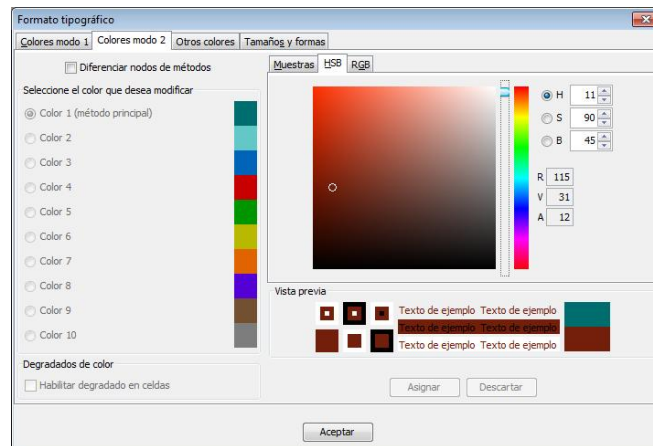


Ilustración 11 Cuadro de formato (segunda pestaña)

La tercera pestaña muestra también un aspecto similar a las dos anteriores, ya que igualmente está orientada a la configuración del color. Esta pestaña permite escoger los colores para los elementos comunes de todas las vistas, tales como el color de las flechas, el fondo de los paneles, etc.

La cuarta de las pestañas, aparece en la Ilustración 12, y está orientada a tamaños, formas y fuentes. Ofrece la posibilidad de elegir un borde para las celdas, el ancho de los marcos para el nodo activo y los nodos que forman parte de la rama actual en expansión, la distancia entre las celdas del árbol recursivo, el tipo de flecha utilizado

o el ancho de la misma. En esta cuarta pestaña también se puede configurar de manera independiente la fuente que se utiliza en las vistas de código y traza en cuanto a tipografía y tamaño (el color se puede graduar en pestañas anteriores).

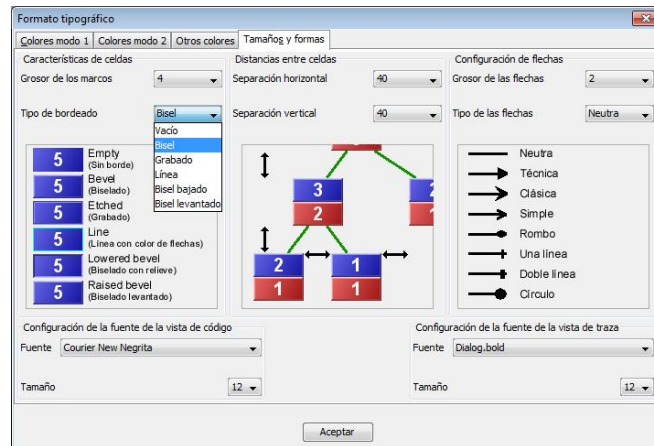


Ilustración 12 Cuadro de formato (cuarta pestaña)

Los cambios van teniendo efecto según se van seleccionando los distintos parámetros configurables y asignando los colores deseados.

Por otra parte, puede configurarse SRec para que los árboles aparezcan compactados y crezcan en anchura gráfica a medida que se van activando las diferentes subllamadas a lo largo de la animación o bien para que abarquen todo el ancho desde el principio. El usuario pueden conmutar entre cada tipo de configuración a través de la opción *Visualización / Árboles compactados*.

4.6 Zoom

La aplicación permite ajustar el tamaño al que se muestran los datos en el árbol recursivo de la visualización y en la pila de control. Para ello, basta activar la opción *Visualización / Configuración de Zoom*.

La aplicación nos ofrecerá la posibilidad de graduar, por separado, el zoom que queremos ejercer sobre la visualización del árbol, la pila de control, la vista de estructura y la cronológica mediante el cuadro de diálogo que aparece en la Ilustración 13.

A medida que se vaya graduando el zoom con la barra deslizante, se podrá ir viendo el tamaño que toman las celdas y los datos que contienen. Si se desea retirar el efecto de zoom sobre la visualización, basta pulsar el botón *Resetear* y los valores de zoom se vuelven 0 para mostrar la visualización en su tamaño y proporción original.

Si no se desea hacer uso de la barra deslizante, se puede escribir el valor deseado de zoom en el cuadro de texto. La aplicación permite además el ajuste manual del zoom mediante los botones *Zoom+* y *Zoom-*, que van ampliando o disminuyendo el zoom con cada pulsación. El botón de ajuste redimensiona el grafo para que se adapte

su parte visible al tamaño actual del panel y se evite así la necesidad de utilizar la barra de desplazamiento. Una vez que el tamaño de las celdas se ajusta a las necesidades, se puede pulsar el botón *Aceptar* para cerrar el cuadro de diálogo.

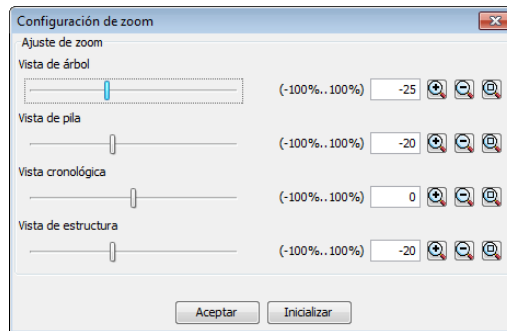


Ilustración 13 Diálogo de gestión de zoom

Además, la aplicación muestra un visor de navegación que permite posicionar la ventana sobre la parte del árbol que se desee. También ofrece información sobre la posición del panel dentro de la extensión del árbol, lo que ayuda a contextualizar la vista.

4.7 Ubicación de los Paneles y las Vistas

SRec permite al usuario ubicar los dos paneles de visualización en dos disposiciones diferentes: uno encima del otro o bien uno al lado del otro. De la primera forma los paneles cobran una proporción horizontal, por lo que determinados árboles de activación o la vista de estructura pueden resultar visualizables muy cómodamente. Por otro lado, cuando los paneles están uno al lado del otro, toman una proporción vertical, lo que suele ser interesante para visualizar la traza de la ejecución, la pila de control, o la vista cronológica de la estructura.

La disposición de los paneles puede configurarse mediante la opción *Visualización / Ubicación de vistas y paneles...*, que también permite ubicar las vistas en los paneles deseados.

Así, el usuario podrá ubicar en cualquiera de los dos paneles cada una de las tres o cinco vistas que esté manejando, con la lógica limitación de que cada panel deberá contener al menos una de las vistas para evitar que quede vacío.

5 Lectura y Edición de Código

Ver de una manera rápida y cómoda el código de cierto algoritmo puede resultar interesante en un momento dado. La aplicación da esta posibilidad a través del panel de código que aparece en el lado izquierdo de la ventana. Para editar tal código, simplemente el usuario tiene que situarse en el punto donde quiera realizar

modificaciones pinchando con el ratón y el cursor le permitirá editar el código. Para que los cambios tengan efecto sobre la visualización, será necesario reprocesar la clase y generar una nueva visualización.

El código aparece en este panel con las líneas numeradas y con la sintaxis coloreada. Este coloreado es configurable desde la opción *Visualización / Formato tipográfico...*

En la parte inferior del panel aparece un área donde SRec informa de los errores que ha detectado la Máquina Virtual de Java al intentar compilar la clase. Si en cualquiera de los mensajes de error que aparecen se selecciona "java:n:" (de manera exacta, dentro de una selección mayor o haciendo triple clic sobre la línea) al soltar el ratón aparecerá seleccionada la línea correspondiente en el área de código, para que así resulte mucho más sencillo encontrar los errores sobre el mismo.

6 Otras Opciones

La aplicación da una serie de opciones de configuración que pueden resultar de interés para el usuario, se indican a continuación.

6.1 Selección de la Máquina Virtual de Java

Esta opción debe ser configurada para dar a la aplicación la capacidad de compilar archivos de código Java y poder generar nuevas visualizaciones a partir de dichos códigos. Si esta opción no es configurada, la aplicación sólo podrá mostrar visualizaciones generadas previamente en otras sesiones cuando sí hubo una máquina virtual disponible.

Para configurar adecuadamente esta opción, se debe seleccionar la opción *Configuración / Máquina virtual Java*. El usuario debe dar la dirección local en la que se encuentra ubicada la máquina virtual de Java 1.6 (es necesario tener instalado el JDK -kit de desarrollo de aplicaciones Java-, no basta con tener el JRE -entorno de ejecución de Java-). En concreto, es necesario proporcionar el path completo del archivo "java.exe", que deberá contar en su mismo directorio con el archivo "javac.exe". Aparte de escribir la dirección manualmente, existe la posibilidad de navegar por el contenido del disco duro mediante el botón *Examinar*.

6.2 Generación de Archivos Intermedios

La aplicación crea durante la etapa de procesado de clases varios archivos que una vez terminado dicho proceso son eliminados por defecto. Sin embargo, la aplicación puede hacer que estos archivos se queden almacenados en el disco duro si así lo desea el usuario. Éste no tiene más que seleccionar las opciones pertinentes en el cuadro de diálogo que le abre SRec.

Para configurar esta posibilidad, es necesario seleccionar la opción *Configuración / Archivos intermedios*. Los ficheros que se generan son:

Archivo con representación XML del código Java original: este archivo contiene una representación en formato XML del código Java que se quiere visualizar.

Archivo con representación XML del código Java generado: este archivo contiene una representación en formato XML del código Java que la aplicación genera para poder almacenar la información necesaria sobre la ejecución del algoritmo deseado.

Archivo Java generado: código Java que incluye sentencias adicionales que permiten el almacenaje de ciertos valores que serán utilizados en la visualización.

Archivo compilado del código Java original: archivo .class del código Java seleccionado, idéntico al generado por el compilador de Java.

Archivo compilado del código Java generado: archivo .class del código Java que genera la aplicación.

Los archivos que más interés pueden tener para los usuarios podrían ser la representación XML del código Java original y el archivo compilado del código Java original (uso del programa como compilador).

6.3 Idioma

SRec permite ser utilizado en múltiples idiomas. Por defecto, la distribución de SRec se ofrece en español e inglés. Configurar el idioma en que SRec muestra los menús, los mensajes, y el resto de textos es muy sencillo.

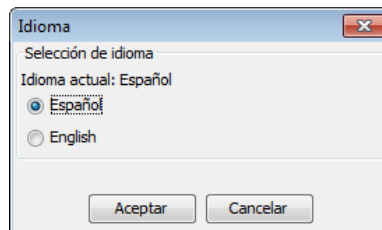


Ilustración 14 Diálogo de gestión de idioma

Basta seleccionar la opción *Configuración / Idioma* para que aparezca un pequeño cuadro de diálogo, mostrado en la Ilustración 14, que deja ver qué idioma está seleccionado en la actualidad. También se puede ver mediante un listado de selección qué otros idiomas están disponibles. Una vez escogido el idioma deseado, se debe pulsar el botón *Aceptar* para que los cambios tengan efecto. En un par de segundos la aplicación actualiza todos los elementos de la ventana con el idioma seleccionado.

6.4 Manejo de Configuraciones

SRec ofrece una serie de opciones que permiten al usuario gestionar distintas configuraciones, las cuales puede usar cuando más le convenga mediante la carga desde el sistema de disco. A continuación se explican las siguientes opciones, accesibles desde el menú *Configuración*:

Restaurar configuración: Recarga la configuración por defecto definida por el usuario. Si hay una visualización abierta en ese instante, los resultados tienen efecto inmediato sobre la misma.

Abrir configuración: Permite cargar una configuración previamente almacenada en disco. Si hay una visualización abierta en ese instante, los resultados tienen efecto inmediato sobre la misma.

Guardar configuración: Guarda la configuración actual en disco para poder ser cargada posteriormente en otra sesión de la aplicación.

Guardar configuración por defecto: Guarda la configuración actual en disco como configuración por defecto. Esta configuración se carga al arrancar la aplicación.

7 Disponibilidad de SRec

SRec ha sido desarrollado por el grupo de investigación LITE (*Laboratory of Information Technologies in Education*), perteneciente al Departamento de Lenguajes y Sistemas Informáticos I, de la Escuela Superior de Ingeniería Informática de la Universidad Rey Juan Carlos.

Puede encontrarse una descripción de la versión 1.0 de SRec en [4], una descripción de la versión 1.1 en [2], información sobre la versión 1.2 en [3] así como de su implementación preliminar en [1]. Puede encontrarse información más completa en la página *web* de la aplicación (<http://www.lite.etsii.urjc.es/srec/>).

Agradecimientos

Este trabajo se ha financiado con el proyecto TIN2008-04103 del Ministerio de Ciencia e Innovación.

Referencias

1. Fernández-Muñoz, L., Pérez-Carrasco, A., Velázquez-Iturbide, J.Á., Urquiza-Fuentes, J.: 2007. A framework for the automatic generation of algorithm animations based on design techniques. In: Duval, E., Klamma, R., Wolpers, M. (eds.): *Creating New Learning Experiences on a Global Scale – EC-TEL 2007*. Lecture Notes in Computer Science, Vol. 4753. Springer-Verlag, Berlin Heidelberg New York (2007) 475–480
2. Velázquez-Iturbide, J. Á., Pérez-Carrasco, A.: Aumentando la Interacción con la Visualización de Algoritmos Recursivos. Proc. X Congreso Internacional de Interacción Persona-Ordenador 2009.
3. Velázquez-Iturbide, J. Á., Pérez-Carrasco, A.: SRec 1.2: visualizador integrado de programas recursivos generales y de Divide y Vencerás. Proc. XI International Symposium on Computers in Education, SIIE 2009.
4. Velázquez-Iturbide, J.Á., Pérez-Carrasco, A., Urquiza-Fuentes, J.: SRec: An animation system of recursion for algorithm courses. Proc. 13th Annual Conf. Innovation and Technology in Computer Science Education, ITiCSE 2008. ACM Press, New York (2008) 225–229