

Antonio Pérez Carrasco
J. Ángel Velázquez Iturbide

Resultado de las tres primeras evaluaciones de usabilidad de SRec

Número 2009-06

Serie de Informes Técnicos DLSI1-URJC
ISSN 1988-8074
Departamento de Lenguajes y Sistemas Informáticos I
Universidad Rey Juan Carlos

Índice

1. Introducción	3
2. Descripción y resultados de las sesiones	4
2.1. Procedimiento general	5
2.2. Resultados de la Sesión 1	5
2.3. Resultados de la Sesión 2	8
2.4. Resultados de la Sesión 3	11
2.5. Resultados globales	13
3. Conclusiones generales	15
Agradecimientos	16
Apéndice	17
A Enunciado y cuestionario de la Sesión 1	17
B Enunciado y cuestionario de la Sesión 2	21
C Enunciado y cuestionario de la Sesión 3	26
D Respuestas íntegras de la Sesión 1	31
E Respuestas íntegras de la Sesión 2	34
F Respuestas íntegras de la Sesión 3	46

Resultados de las tres primeras evaluaciones de usabilidad de SRec

Antonio Pérez Carrasco¹, J. Ángel Velázquez Iturbide¹
[S1]

¹ LITE – Laboratory of Information Technologies in Education
Departamento de Lenguajes y Sistemas Informáticos I
Escuela Técnica Superior de Ingeniería Informática
Universidad Rey Juan Carlos, Madrid, España
{ antonio.perez.carrasco, angel.velazquez } @urjc.es

Abstract. Se presentan las tres primeras sesiones de evaluación de usabilidad realizadas sobre SRec, una herramienta software con fines docentes orientada tanto a profesores como alumnos. Se persigue con estas sesiones por un lado evaluar el desarrollo realizado y por otro obtener información sobre qué funciones son más y menos útiles para el alumnado, al cual la aplicación intenta ayudar en el estudio de los algoritmos en función de la técnica de diseño empleada para desarrollarlos.

Keywords: Visualización, Algoritmo, Recursividad, Animación.

1 Introducción

SRec es una herramienta software creada por el grupo de investigación LITE que comenzó su desarrollo el 11 de septiembre de 2006 tras un periodo de investigación y definición del prototipo inicial. Su objetivo es mitigar el esfuerzo necesario para la generación de visualizaciones de programas basados en la recursividad aportando vistas específicas en función de la técnica de diseño empleada.

Este software está orientado a los profesores, que encontrarán en SRec una herramienta que les facilitará la creación sin esfuerzo de numerosos ejemplos para sus clases. Además, también está enfocado a los alumnos, ya que les permitirá entender mejor el funcionamiento de los algoritmos replicando los ejemplos vistos en clase y creando otros nuevos, así como desarrollando prácticas y resolviendo ejercicios.

Su desarrollo ha estado ligado desde el principio a las necesidades encontradas durante la actividad docente, por lo que es fruto, tal y como está concebido, de un cuidado proceso de estudio que ha intentado dotar a la aplicación de las facilidades y funcionalidades que de mayor utilidad puedan resultar a los futuros usuarios (profesores y alumnos).

SRec ha sido empleado en clases magistrales desde el curso 2007/2008 para apoyar al profesor en sus exposiciones ante el alumnado. Éste a su vez también ha tenido oportunidad de utilizar el software para la realización de prácticas y repaso de ejemplos y ejercicios propuestos.

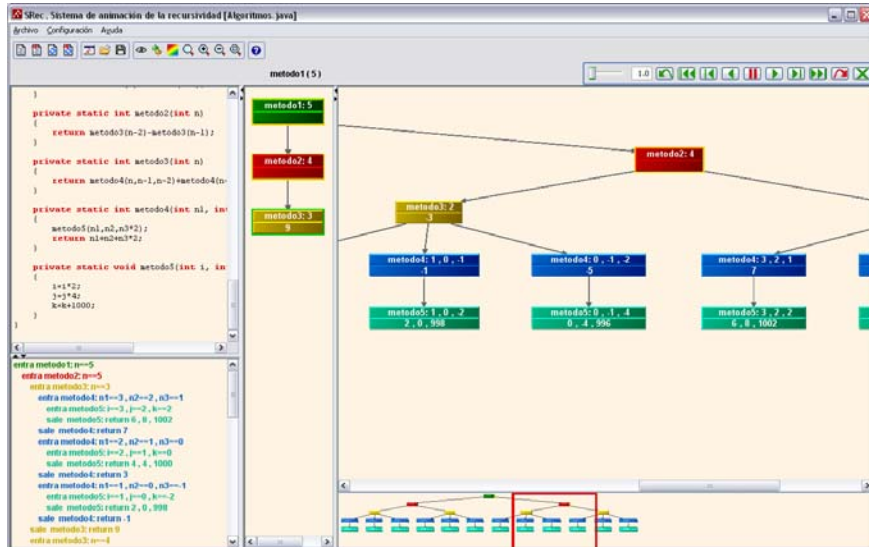


Fig. 1. Vista principal de SRec, con una animación mostrada en un estado intermedio

2 Descripción y resultados de las sesiones de evaluación

A lo largo de los cursos 2006/2007 a 2008/2009 se han realizado tres evaluaciones de usabilidad entre diferentes grupos de alumnos para realizar una medición tanto del grado de aceptación de SRec y de las animaciones que produce como de los cambios registrados en los resultados académicos del alumnado tras su comienzo de utilización.

Con el fin de medir la calidad de la aplicación y de sus diferentes cualidades, se han llevado a cabo tres sesiones de evaluación de usabilidad para determinar si se estaba caminando en la dirección correcta o si por el contrario era necesario corregir el rumbo de la aplicación. Las sesiones de evaluación de usabilidad han servido, por tanto, para refutar ciertas funcionalidades o modos de funcionamiento de la aplicación así como para recabar ideas que posiblemente pudieran materializarse después.

Estas sesiones de evaluación se han realizado siempre entre alumnos y los ejercicios propuestos en las sesiones han estado siempre contenidos en alguna de las prácticas que formaban parte de la actividad académica de las asignaturas por lo que los alumnos tenían la motivación adecuada para afrontar los ejercicios propuestos y aprender a manejar la aplicación. Salvo en la primera de ellas, la sesión de evaluación se realizó después de que los alumnos vieran el uso de la aplicación gracias al empleo de la misma por parte del profesor en clases previas.

2.1 Procedimiento general

Estas sesiones tuvieron una duración de dos horas. En ellas se plantearon varios ejercicios relacionados con la programación y análisis de algoritmos. El primero de ellos siempre fue realizado por el profesor con el fin de presentar la aplicación e introducir al alumnado al manejo básico de la misma. Los siguientes ejercicios, de baja complejidad, se enfocaban en que el alumno se familiarizase con la aplicación mediante la utilización de sus diversas funcionalidades para que pudieran abordar la práctica de una manera eficiente.

Posteriormente se les planteaba un último ejercicio que les comenzaba a exigir cierta capacidad de análisis y un mínimo nivel de destreza en el diseño de algoritmos. Este ejercicio, que debía efectuarse con la herramienta, era objeto de evaluación para la asignatura.

Por último, se les facilitaba un breve cuestionario que pedía su valoración en una escala de 1 a 5 para ciertos aspectos concretos de la aplicación. Además, el cuestionario planteaba preguntas de respuesta abierta. Tanto los enunciados como las respuestas de todos los participantes se pueden consultar en los anexos de este informe.

2.2 Resultados de la Sesión 1

2.2.1 Procedimiento de la sesión

La primera de las tres sesiones de evaluación de la usabilidad tuvo lugar el 24 de mayo de 2007. Se contó para ella con la participación de 7 alumnos de la asignatura “Estructura de datos y algoritmos avanzados”, impartida por D. Carlos A. Lázaro Carrascosa.

Se le planteó como ejercicio final de la práctica la depuración de una versión del algoritmo mergesort que contenía dos errores.

2.2.2 Resultados de satisfacción

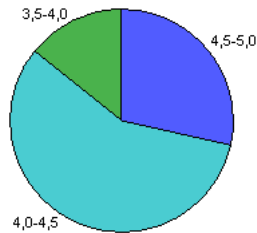
Los resultados de la prueba permitían obtener la conclusión de que el programa necesitaba pequeñas mejoras en la interfaz. Así, algunos alumnos no fueron conscientes de algunas de las opciones que proporcionaba el programa como por ejemplo la edición de código, utilizando otro software ajeno para esa tarea. Al fin y al cabo, SRec no está orientado a la depuración de programas, sino a la visualización de algoritmos ya programados, si bien es cierto que la integración de su sencillo editor permite agilizar las pequeñas depuraciones que puedan ser necesarias en un momento dado. SRec era además poco flexible en algunos momentos, comportamiento que fue corregido para evitar (o al menos reducir) los mensajes de error dados al usuario por parte de la aplicación.

Como nota global, SRec obtuvo un 4,14 sobre 5. Respecto a usabilidad, SRec obtuvo un buen 4,43 sobre 5. A continuación se presenta una tabla con los valores medios obtenidos en los cuestionarios:

Tabla 1. Valores obtenidos en el cuestionario de la sesión 1

Cuestión	Media
SRec es fácil de usar	3,88
SRec me ha ayudado a analizar algoritmos y a encontrar el error	2,63
SRec me ha ayudado a analizar algoritmos para comprobar que la solución propuesta es correcta	4,13
Calidad general de SRec para analizar la recursividad	3,38
Calidad del menú principal	3,75
Calidad de los controles de animación	4,38
Calidad de la vista de traza	3,75
Calidad de la vista de la pila de control	4,00
Calidad de la vista del árbol de recursión	4,25
Calidad de la configuración de las visualizaciones	3,88
Interacción con los paneles	3,63
SRec me ha gustado	3,63

Atendiendo a las medias individuales de las puntuaciones aportadas por cada alumno, se muestra en la figura 2 la valoración obtenida de SRec (sobre 5):

**Fig. 2.** Valor de la media de las puntuaciones aportadas por cada individuo en la sesión 1

Tal y como se puede ver en el gráfico, la primera versión evaluada de SRec, extremadamente sencilla en comparación con la versión actual, obtuvo una gran aceptación. Así, más del 28% de los alumnos puntuaron a SRec con una nota media superior al 4,5 sobre 5, mientras que el 57% le situó entre el 4 y el 4,5.

2.2.3 Críticas y sugerencias recibidas

En términos generales, como ya se ha comentado, la aceptación del programa fue bastante alta. La característica que más gustó a los alumnos fue la facilidad del manejo de la herramienta. El hecho de que con apenas un par de clics se puedan generar visualizaciones útiles y con calidad gráfica fue uno de los factores que más ayudó a lograr la aceptación de la herramienta entre los alumnos.

La vista del árbol de recursión, que les permitía ver de una manera clara cómo funcionaba el algoritmo que estaban visualizando, fue el segundo elemento que más impacto positivo tuvo en los alumnos. Por detrás quedan otros factores como la claridad y utilidad de las visualizaciones en general y la capacidad de personalización de las vistas que ofrece la aplicación.

Algunos alumnos también valoraron positivamente las vistas de la traza y de la pila de control, que complementaban la vista principal. También hubo quien destacó que se podía ver fácilmente el resultado de cada subllamada y de la ejecución total del algoritmo, lo cual le resultaba de interés.

Respecto a las opiniones negativas sobre SRec, las más destacadas son las que echan en falta el seguimiento de la ejecución a través del panel de código, remarcando las sentencias que se van ejecutando. Sin embargo, la aplicación no funciona por sentencias, sino por subllamadas, por lo que el enfoque del resaltado de sentencias no se adapta al funcionamiento del programa. La inclusión de valores de variables en la vista de traza sigue un planteamiento similar y requeriría igualmente que la aplicación avanzase por sentencias, en lugar de por subllamadas recursivas como es su funcionamiento.

Para otros alumnos, las principales quejas fueron la dificultad que presentaba ver árboles de gran tamaño, ya que se perdían por él. Posteriormente se implementaría un visor de navegación para ofrecer una vista contextual de todo el árbol que permita al alumno ubicarse dentro del árbol y navegar por él con mayor facilidad.

También se destacó que el programa no ayudaba activamente a encontrar los errores que contenían los códigos suministrados en el enunciado de la práctica. Esto es fruto de que la aplicación no ha sido concebida como un depurador sino como una herramienta para el análisis de algoritmos correctos.

Por último, algunos alumnos indicaron que el funcionamiento del programa podría ser más cómodo para el usuario si incluyese un botón de recarga de una clase para cuando ésta ha sido modificada por el usuario. De esta forma, se evita tener que buscarla por el sistema de ficheros cada vez que se quiere volver a cargar. El botón fue añadido posteriormente para agilizar la carga de clases, y se acompañó de otro que facilitaba la propia edición de la clase ya cargada en el sistema.

A la pregunta que pedía la mención de características que SRec no tenía pero que sería interesante que incorporara, casi la mitad de las respuestas solicitaron el resalte dinámico de instrucciones de código a medida que se iba avanzando paso a paso en la visualización. Otras peticiones fueron el resalte de ciertos nodos (implementado posteriormente en la versión 1.1), más opciones en la vista de traza, resalte de errores sintácticos en el sencillo editor que proporciona SRec, la modificación de las combinaciones de teclado para el acceso rápido a las funcionalidades de SRec o más opciones en el desarrollo de visualizaciones (si bien no se especificó ninguna funcionalidad concreta).

2.2.4 Cuestiones académicas

Tan sólo tres estudiantes de los siete que realizaron la prueba resolvieron correctamente el ejercicio. Se debe reconocer que SRec no fue concebido como un depurador de algoritmos, sino más bien como un asistente al análisis, de ahí que no ofrezca muchas facilidades para realizar tal tarea, salvo el editor integrado de código.

2.3 Resultados de la Sesión 2

2.3.1 Procedimiento de la sesión

La segunda prueba fue desarrollada el 4 de diciembre de 2007 dentro de la asignatura “Diseño y análisis de algoritmos”, impartida por D. Ángel Velázquez Iturbide. Fue de carácter similar, y con una intencionalidad igualmente parecida a la de la primera prueba.

Se les propuso a los 28 alumnos asistentes a la sesión una serie de ejercicios que tenían que realizar. Como ejercicio final se le pidió a los alumnos que analizaran el algoritmo de recursividad múltiple del problema de competición para entregar posteriormente dos grafos: uno del árbol de recursión (copiado desde la pantalla de SRec) y otro del grafo de dependencia.

2.3.2 Resultados

Los resultados de la prueba, que se hicieron sobre una versión más avanzada del programa y que tenía en cuenta los resultados de la prueba anterior, tuvieron un carácter muy positivo.

Las mejores opiniones sobre el programa se vertieron sobre la vista del árbol de activación, gracias a su diseño estilizado y altamente configurable y a que ayuda a entender fácilmente el proceso recursivo, tal y como indicó un notable número de alumnos. Esta vista es, por su expresividad, calidad técnica y vía de información, uno de los elementos más valorados de la aplicación. En general, SRec obtiene mejores puntuaciones que en la primera sesión, fruto de ciertas mejoras en el procesamiento de archivos y opciones de funcionamiento, que hacen que su funcionamiento sea más fluido y estable.

A continuación se adjunta una tabla que repasa las puntuaciones obtenidas por el programa:

Tabla 2. Valores obtenidos en el cuestionario de la sesión 2

Cuestión	Media
SRec es fácil de usar	4,50
SRec me ha ayudado a analizar algoritmos para estudiar qué llamadas se realizan en tiempo de ejecución	4,29
SRec me ha ayudado a identificar la dependencia de llamadas recursivas	4,36
Calidad general de SRec para analizar la recursividad	4,29
Calidad del menú principal	4,07
Iconos	3,86
Calidad de los controles de animación	4,50
Calidad de la vista de traza	4,00
Calidad de la vista de la pila de control	4,04
Calidad de la vista del árbol de recursión	4,43
Calidad de la configuración de las visualizaciones	3,82
Interacción con los paneles	3,89
SRec me ha gustado	4,26

De nuevo valoramos también la nota media obtenida por el programa (siempre sobre 5) otorgada por cada uno de los participantes, donde se observa que aparece un nuevo grupo:

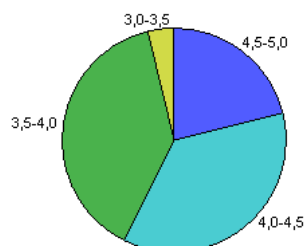


Fig. 3. Valor de la media de las puntuaciones aportadas por cada individuo en la sesión 2

El porcentaje de estudiantes que puntuaron a SRec con una nota mayor a 4,5 alcanza el 21% (7 puntos menos que en la anterior sesión), mientras que el de alumnos que puntuaron a SRec con entre 4 y 4,5 puntos supera el 37%, haciéndose significativo (y mayoritario) el aumento del grupo de alumnos que puntúa a SRec con una media situada en entre 3,5 y 4 puntos, que asciende al 39%.

Estos datos contrastan con los extraídos desde las notas medias de las diferentes cuestiones planteadas en el formulario, donde en general, SRec obtenía unos resultados notablemente mejores que en la primera sesión de evaluación de su usabilidad.

2.3.3 Críticas y sugerencias recibidas

En esta ocasión la herramienta también consiguió un alto éxito de aceptación entre los alumnos, si bien el mayor número de participantes en la sesión proporcionó una mayor variedad de opiniones y sugerencias que en la anterior sesión.

La cualidad más resaltada por los estudiantes fue la claridad y utilidad de las vistas proporcionadas por SRec. Así, dejaban claro que las visualizaciones realmente les ayudaba en el proceso de comprensión y análisis del funcionamiento de los algoritmos, objetivo fundamental del programa. La vista del árbol de recursión fue también muy bien recibida por los alumnos, siendo el segundo elemento del programa mejor aceptado. Dos tercios de los individuos encuestados respaldaron una de estas dos opiniones expuestas.

A continuación, la vista de la pila de control se sitúa como el siguiente elemento con mayor aceptación, vista que complementa la información mostrada en el árbol de recursión. Con un nivel de aprobación similar se encuentran las opciones de manejo del programa, calificadas como sencillas por gran parte del alumnado.

Ya en menor medida, también contaron con cierto respaldo por parte de los alumnos la vista de traza y la capacidad de aprendizaje que proporciona SRec gracias a la visión conjunta de las vistas proporcionadas, todas ellas complementarias y siempre con un funcionamiento sincronizado. También recibieron mención la flexibilidad de configuración de la aplicación y la funcionalidad de edición de código.

En el lado negativo, las funcionalidades que más críticas recibieron fueron la exportación de una visualización a GIF animado y la vista de traza. El proceso de

exportación a GIF animado exigía en esta versión mantener la ventana visible mientras se realizaba el guardado de la animación, un proceso que, en visualizaciones grandes, podía tomar varios minutos, de tal forma que no se podía emplear el ordenador para otras tareas de manera concurrente. Este proceso fue optimizado en siguientes versiones, eliminando esa restricción. Por otro lado, varios alumnos expresaron un ligero rechazo a la vista de traza, fundamentalmente por considerarla redundante.

De nuevo varios alumnos volvieron a mencionar la dificultad que podía llegar a tener la visualización de árboles muy grandes, ya que por entonces aún se contaba con una versión 1.0 preliminar del programa, que no contenía ningún visor de navegación. También quedó mencionado que la aplicación podía tardar algunos segundos en abrir una visualización con un árbol de recursión muy grande. Otras cuestiones objeto de crítica fueron el alto número de opciones de configuración disponibles, la poca flexibilidad de la barra de herramientas, la no existencia de libre movilidad de los paneles que contenían las vistas y el hecho de que el panel de código redujera el espacio disponible que se podía destinar a las vistas.

Como principales sugerencias recibidas, destaca de nuevo la petición de incluir en la vista del código de la clase el resaltado de las instrucciones que se van ejecutando. También destacaron las peticiones de que se admitieran otros lenguajes de entrada para la codificación de los algoritmos como C, si bien el lenguaje con el que se trabaja en la asignatura es Java. También, fruto de los requisitos del enunciado, nació una nueva demanda, la de la representación del grafo de dependencias, para que complementara las vistas existentes.

Por entonces, la aplicación sólo podía exportar animaciones GIF, de ahí que ante la petición en el enunciado de que aportaran en su informe de prácticas un árbol de recursión, pidieran que la aplicación pudiera realizar capturas estáticas de la vista del árbol de recursión en formato JPG. Esta funcionalidad fue añadida posteriormente, permitiendo la exportación estática de las vistas en archivos GIF, JPG y PNG, empleables en documentación de carácter ofimático.

Otras sugerencias recibidas fueron la modificación de atajos de teclado, la funcionalidad de autozoom según va avanzando la animación de la visualización, mayores opciones de movilidad de los paneles, la creación de un visor de navegación (se creó poco después), información sobre si la ejecución del algoritmo levantó alguna excepción (se implementó posteriormente), cálculo de la complejidad del algoritmo en tiempo y memoria (actualmente la aplicación ofrece el número total de subllamadas realizadas), opción directa para imprimir en papel el contenido de las vistas, aportación como nueva vista de un diagrama de flujo, inserción en el árbol de la operación que realiza el algoritmo y la posibilidad de “autoformato” en la vista del árbol de recursión.

2.3.4 Cuestiones académicas

Respecto a los resultados académicos de las pruebas, 26 de los 28 alumnos entregaron correctamente el ejercicio, y de éstos, 23 emplearon SRec para la resolución del mismo. En esta ocasión, la nota sobre usabilidad subió hasta el 4,5 sobre 5, mientras que la nota final del programa alcanzó los 4,26 puntos sobre 5.

2.4 Resultados de la Sesión 3

2.4.1 Procedimiento de la sesión

La tercera sesión de evaluación de usabilidad se desarrolló el 14 de noviembre de 2008. De nuevo los participantes, 33 en total, eran alumnos de la asignatura “Diseño y análisis de algoritmos”, impartida por D. Ángel Velázquez Iturbide. Se mantuvo la estructura de la sesión, por lo que fue muy parecida a la de las pruebas anteriores.

Tras los ejercicios de familiarización con la herramienta, en esta tercera sesión se les planteó un problema donde tenían que eliminar la redundancia debida a la recursividad múltiple empleando técnicas de memorización y tabulación, ofreciendo posteriormente además un cálculo de la complejidad de los algoritmos resultantes. Adicionalmente tenían que dibujar un árbol de recursión (la aplicación les permitía sacar una imagen capturada del mismo) y su grafo de dependencia asociado.

2.4.2 Resultados

A continuación se adjuntan los datos medios (sobre 5) obtenidos en las preguntas realizadas en el cuestionario de la tercera sesión.

Tabla 3. Valores obtenidos en el cuestionario de la sesión 3

Cuestión	Media
SRec es fácil de usar	4,20
SRec me ha ayudado a analizar algoritmos y determinar las llamadas recursivas que se realizan	4,19
SRec me ha ayudado a analizar algoritmos para identificar la dependencia de llamadas recursivas	4,05
Calidad general de SRec para analizar la recursividad	4,00
Iconos	3,57
Calidad de los controles de animación	3,71
Calidad de la vista del árbol de recursión	4,00
Calidad del visor de los árboles grandes	3,86
Configuración de formatos	3,76
Configuración de zoom	3,71
Interacción con los paneles	3,80
Proceso de generación de una animación	4,00
Proceso de almacenamiento/carga de una animación	4,14
Visualización almacenada en un fichero de captura	4,10
SRec me ha gustado	3,95

Los resultados, tal y como se puede observar, muestran un ligero descenso en los valores, fruto de la mayor complejidad de la aplicación y de algunos problemas de instalación que surgieron debido a una incorrecta configuración de los equipos del laboratorio el día en que se llevó a cabo la sesión.

De manera individual, los resultados obtenidos por SRec se muestran en el gráfico de la figura 3.

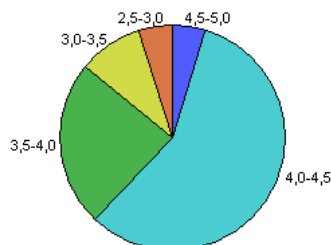


Fig. 4. Valor de la media de las puntuaciones aportadas por cada individuo en la sesión 3

A pesar de que se mantiene la tendencia decreciente del grupo que puntúa a SRec con más de 4,5 puntos, que se reduce al casi 5%, la proporción total de personas que puntuaron a SRec con una nota superior a 4 ascendió hasta el 62% (en la anterior sesión se mantuvo en el 58%), gracias a que más del 57% de los alumnos colocó a SRec una nota media de entre el 4 y el 4,5.

Casi el 24% de los encuestados dio como nota media de SRec un valor entre 3,5 y 4 puntos, mientras que sólo el 14% restante dio una nota por debajo del 3,5, y siempre superior al 2,5 (nivel de aprobado).

2.4.3 Críticas y sugerencias recibidas

En esta ocasión la cantidad de respuestas abiertas contestadas fue menor, tanto en aquellas que pedían valorar los aspectos positivos como en las de los aspectos negativos. Aun así, se pudo recoger información de interés sobre los puntos fuertes de la aplicación o las carencias que los alumnos detectaron en SRec.

Al igual que en la segunda sesión, la principal cualidad de SRec destacada por los alumnos fue la alta claridad y utilidad de las vistas ofrecidas, haciendo de la aplicación una herramienta de calidad para su periodo de aprendizaje. La facilidad de manejo, una cualidad que se intentó mantener a toda costa pese a la cada vez mayor cantidad de funcionalidades, y la vista del árbol de recursión fueron nuevamente dos de las cualidades mejor valoradas por los alumnos.

Otros elementos de SRec que fueron mencionados positivamente son: la vista de traza, la herramienta de zoom, la capacidad de exportación a formato GIF animado, el proceso de carga de clases, y la posibilidad de guardar las visualizaciones en formato XML para recuperarlas en sesiones posteriores.

En el polo opuesto, la ingente cantidad de opciones de configuración pareció no agrandar a algunos alumnos, que destacaron esto como un aspecto negativo, bien por considerarlo irrelevante o demasiado complejo.

El proceso de exportación a GIF animado (por el hecho de no ser instantáneo y no permitir su cancelación) recibió alguna crítica, igual que la visión de árboles grandes, que pese a contar ya con un visor de navegación, algunos alumnos pudieron experimentar alguna dificultad.

Otras críticas se enfocaron en el tiempo de ejecución y creación de visualizaciones en el caso de árboles grandes, la barra de herramientas por su poca flexibilidad de ubicación en la ventana, el proceso de carga de clases y la interfaz general, a la que se proponía realizar cambios sin especificar ninguno.

Como principal reivindicación apareció la generación de un grafo de dependencias, comprensible al ser una representación requerida en el ejercicio evaluable que realizaron durante la sesión y que no les aportaba el programa. La segunda petición fue la de un cambio en los iconos que representan las diferentes funcionalidades de SRec tanto en los menús como en la barra de herramientas, al parecer crean confusión y resultan poco intuitivos.

Otras peticiones, menos secundadas que las anteriores, fueron la de que SRec se pueda ejecutar en otras plataformas distintas a Windows, la adición de nuevas opciones de visualización, como la de pantalla completa o la exportación de capturas a gran tamaño.

2.4.4 Cuestiones académicas

Las notas fueron considerablemente mejores que las obtenidas en prácticas anteriores de similares características, logrando duplicar el número de alumnos que obtuvieron la calificación de “Sobresaliente”. Espontáneamente la mayoría de los alumnos indicaron en sus informes de prácticas que la herramienta les había ayudado a entender mejor el funcionamiento del algoritmo estudiado y les parecía de ayuda la aplicación.

2.5 Resultados globales

Los resultados de las diferentes sesiones dejan ver que el programa, poco a poco, ha ido satisfaciendo cada vez en mayor profundidad las necesidades del profesorado y del alumnado.

A medida que las evaluaciones han ido produciéndose, se ha ido obteniendo una realimentación cada vez más fina, y que ha ido pasando de reclamar mejoras básicas para pasar a requerir funcionalidades más específicas.

Es cierto que la aplicación ha ido ganando en complejidad, sin embargo, los alumnos en todo momento han percibido la herramienta como una aplicación fácil de aprender a manejar, con unos controles intuitivos y con una clara e incontestable utilidad, asistir al alumno durante el proceso de aprendizaje de algoritmos recursivos. Ellos mismos lo afirmaron tanto en los cuestionarios que se les pasó durante las diferentes sesiones de evaluación de usabilidad como en los informes de prácticas presentados con posterioridad, donde no se les pedía ya opinión alguna sobre la aplicación.

Los resultados de aceptación parecen evidenciar que la utilización de herramientas software motiva en mayor grado al alumno durante el proceso de aprendizaje que la resolución tradicional de los ejercicios sin ayuda de ellas.

Para facilitar la comparativa, se presenta a continuación la tabla conjunta de valoración con las respuestas a las cuestiones de las tres sesiones de evaluación de usabilidad realizadas hasta el momento. Tal y como se puede comprobar, las preguntas del cuestionario fueron variando a lo largo de las sesiones.

Tabla 4. Valores medios obtenidos en los tres cuestionarios

Cuestión	Sesión 1	Sesión 2	Sesión 3
SRec es fácil de usar	3,88	4,50	4,20
SRec me ha ayudado a analizar los algoritmos recursivos para buscar el error	2,63	-	-
SRec me ha ayudado a comprobar que la solución propuesta es la correcta	4,13	-	-
SRec me ha ayudado a analizar algoritmos y a determinar las llamadas recursivas que se realizan	-	4,29	4,19
SRec me ha ayudado a analizar algoritmos para identificar la dependencia de llamadas recursivas	-	4,36	4,05
Calidad general de SRec para analizar la recursividad	3,38	4,29	4,00
Estructura del menú principal	3,75	4,07	-
Controles de animación	4,38	4,50	3,71
Vista de traza	3,75	4,00	-
Vista de pila de control	4,00	4,04	-
Vista de árbol de activación	4,25	4,43	4,00
Iconos	-	3,86	3,57
Calidad del visor de los árboles grandes	-	-	3,86
Configuración de formatos	3,88	3,82	3,76
Configuración de zoom	-	-	3,71
Interacción con los paneles	3,63	3,89	3,80
Proceso de generación de una animación	-	-	4,00
Proceso de almacenamiento/carga de una animación	-	-	4,14
Visualización almacenada en un fichero de captura	-	-	4,10
SRec me ha gustado	3,63	4,26	3,95

Como se puede apreciar, la segunda sesión ofrece los mejores resultados en los cuestionarios, ya que los resultados de la tercera sesión son ligeramente inferiores. Esto puede deberse a ciertos problemas existentes en el aula a la hora de ejecutar la aplicación, que provocaron improvisar algunas soluciones de emergencia, y a una mayor complejidad de la aplicación, fruto de las funcionalidades añadidas a versiones anteriores más sencillas.

Desde el punto de vista de la satisfacción personal, los valores obtenidos son los siguientes:

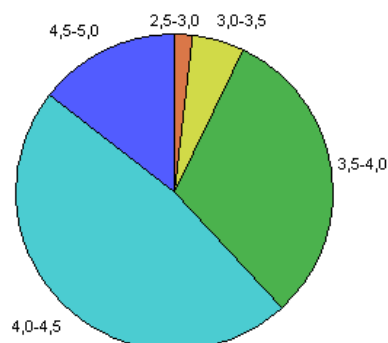


Fig. 5. Valor de la media de las puntuaciones aportadas por cada individuo en las tres sesiones

Los datos reflejan la excelente aceptación de la aplicación entre los usuarios. Más de un 61% le otorga una nota superior a 4 (sobre un total de 5), siendo el más numeroso de los grupos el que le concede una puntuación en el rango (4,4.5], que asciende al 47,27% de los encuestados, casi la mitad.

El segundo grupo más numeroso, con un 30,9% de los encuestados, es el que proporciona a SRec una nota entre 3,5 y 4, éste inclusive. Los grupos que dan a la aplicación una nota inferior de 3,5 tienen un carácter residual, ya que juntos no superan el 7,2%. Todos los encuestados ofrecen una media que permite alcanzar el umbral del aprobado a SRec en cuanto a aceptación y, como ya se ha comentado, alcanzar el notable alto o sobresaliente en el 61% de los casos.

3 Conclusiones generales

Las sesiones de usabilidad se realizaron separadas en el tiempo y sobre versiones de SRec cada vez más completas y complejas. Estas versiones eran el resultado de largos procesos de diseño e implementación que iban acercando la aplicación hacia el resultado final esperado, una herramienta fácil de aprender, fácil de usar, que asista al alumno en su proceso de aprendizaje y que ayude al profesor a complementar sus exposiciones teóricas. A medida que se van ampliando las funcionalidades y posibilidades de la herramienta se requiere de un mayor esfuerzo por parte del usuario a la hora de aprender el manejo de la herramienta, aunque este aspecto ha sido tenido en cuenta de manera permanente durante el diseño de la aplicación. Así, destaca el dato de que la pregunta con mayor puntuación del cuestionario de las sesiones 2 y 3 es la que refleja si SRec resulta fácil de usar, lo que indica que este aspecto, de carácter fundamental, ha sido conseguido.

En general, el alumnado ha sido muy receptivo a SRec, encontrando una herramienta de apoyo que le ha ayudado en la consecución de buenos resultados académicos gracias a las animaciones, que permiten realizar un seguimiento paso a paso muy visual de la ejecución de los algoritmos, motivando e involucrando activamente al alumno en el proceso de aprendizaje.

Agradecimientos

Este trabajo se ha financiado con el proyecto TIN2008-04301 del Ministerio de Innovación y Ciencia. Agradecemos a Asunción Aldave Izaguirre la tabulación en Excel de las respuestas del cuestionario recogidas en papel.

A Enunciado y cuestionario de la sesión 1

SRec es una aplicación para la visualización y animación de métodos recursivos. Esta sesión de laboratorio se realiza para evaluar la calidad de SRec. La sesión consta de varias fases a realizar secuencialmente, que se describen a continuación.

1. **Demostración del profesor.** Duración: 10-15 minutos. Algoritmo: serie de Fibonacci recursiva.

```
public static int fib (int n) {
    if (n==0 || n==1)
        return 1;
    else
        return fib(n-1) + fib(n-2);
}
```

El profesor realizará una demostración del funcionamiento de SRec en la que mostrará el funcionamiento de sus funciones principales: abrir ficheros, almacenar/cargar una animación, reproducir la animación, manejar paneles y cambiar opciones de configuración.

2. **Primera animación.** Duración aproximada: 20 minutos. Algoritmo: exponenciación mediante partición binaria del exponente.

```
public static int pot (int b, int e) {
    if (e==0)
        return 1;
    else if (e%2==0)
        return pot(b*b,e/2);
    else
        return b*pot(b*b,e/2);
}
```

El alumno generará una animación de este algoritmo. Para ello, debe cargar la clase *ClaseEvaluacion* disponible en la web de la asignatura, generar una animación, almacenarla y cargarla de nuevo.

3. **Segunda animación.** Duración aproximada: 20 minutos. Algoritmo: números combinatorios.

```
public static int comb (int m, int n) {
    if (n==0)
        return 1;
    else if (m==n)
        return 1;
    else
        return comb(m-1,n) + comb(m-1,n-1);
}
```

De nuevo, el alumno experimentará con una animación de un algoritmo recursivo múltiple. Se le pide que genere una animación y, después, experimente con las múltiples vistas y con las opciones de configuración.

4. **Tercera animación.** Duración aproximada: 40 minutos. Algoritmo: ordenación por mezcla.

```
private static void ordenarPorMezcla2
    (int[] v, int inf, int sup) {
    if (inf<sup) {
        int medio = (inf+sup)/2;
        ordenarPorMezcla2 (v, inf, medio);
        ordenarPorMezcla2 (v, medio, sup);
        mezclar2 (v, inf, medio, sup);
    }
}

public static void mezclar2
    (int[] v, int inf, int medio, int sup){
    int[] vAux = new int[sup-inf+1];
        //vector auxiliar de igual long
    int i1 = inf;
    int i2 = medio+1;
    int j = 0; //inicialización índice vector aux.
    while (i1<=medio && i2<=sup) {
        if (v[i1]>v[i2]) {
            vAux[j] = v[i1];
            i1++;
        }
        else {
            vAux[j] = v[i2];
            i2++;
        }
        j++;
    }
    for (int i=i1; i<=medio; i++) {
        vAux[j] = v[i];
        j++;
    }
    for (int i=i2; i<=sup; i++) {
        vAux[j] = v[i];
        j++;
    }
    for (int i=inf; i<=sup; i++)
        v[i] = vAux[i-inf];
        //transformación de índices y copia
    }
}
```

El alumno debe depurar este programa que implementa el algoritmo de ordenación por mezcla. Puede servirse de SRec o estudiar el código de forma

estática. Debe entregar un fichero (nombrado con los apellidos del alumno) en el que especifique los números de línea que contienen errores, cada fragmento de código erróneo y cada nuevo fragmento.

5. **Cuestionario.** Duración aproximada: 20 minutos.
El alumno debe responder a las preguntas siguientes.

Cuestionario

En las dos preguntas siguientes, marca la opción de cada pregunta con la que estás de acuerdo.

1. Sobre la facilidad de uso de SRec:
 - La herramienta es fácil de usar
 - La herramienta es fácil de usar en unas partes pero difícil en otras.
Identifique qué partes: _____
 - La herramienta es difícil de usar
2. Sobre la utilidad de SRec:
 - La herramienta es muy o bastante útil
 - La herramienta es algo útil
 - La herramienta es poco o nada útil

En las tres preguntas siguientes, evalúa cada opción con un número comprendido entre 1 (muy malo) y 5 (muy bueno).

3. Evalúa la calidad de las siguientes características de SRec:
 - Estructura del menú principal
 - Facilidades de almacenar/cargar animación
 - Facilidades de configuración de las visualizaciones
 - Controles de animación
 - Múltiples vistas de la recursividad
4. Evalúa la utilidad de las siguientes características de SRec:
 - Facilidades de almacenar/cargar animación
 - Facilidades de configuración de las visualizaciones
 - Controles de animación
 - Múltiples vistas de la recursividad
5. Evalúa la utilidad de las siguientes representaciones de la recursividad:
 - Traza
 - Pila de control
 - Árbol de activación
6. Evalúa la utilidad de SRec para depurar el algoritmo de ordenación por mezcla: ____
7. Evalúa la utilidad de las siguientes representaciones de la recursividad para depurar:
 - Traza
 - Pila de control
 - Árbol de activación

8. Evalúa cuánto te ha gustado SRec: ____

Responde a las siguientes preguntas en formato libre:

9. Di qué características te parece que serían útiles pero SRec carece de ellas:

10. Di qué características de SRec te parecen tan poco útiles que las suprimirías:

11. Describe las ventajas que encuentras en SRec:

12. Describe los inconvenientes que encuentras en SRec:

B Enunciado y cuestionario de la Sesión 2

Ingeniería Informática Asignatura *Diseño y Análisis de Algoritmos*

Curso 2007/2008 Práctica nº 2

Objetivo

El objetivo de la práctica es que el alumno practique en la eliminación de la recursividad múltiple.

Carácter

La práctica es voluntaria. La sesión en el laboratorio se realizará individualmente. El resto de la práctica puede realizarse en parejas.

Prerrequisitos

El alumno debe tener nociones básicas de eliminación de la recursividad múltiple.

Enunciado

SRec es una aplicación para la visualización y animación de métodos recursivos. La práctica consta de dos partes, una a realizar en el laboratorio y otra, fuera del mismo.

La sesión de laboratorio también sirve para evaluar la calidad de SRec. La sesión consta de cuatro fases a realizar secuencialmente:

1. **Demostración del profesor.** Duración aproximada: 15 minutos. Algoritmo: serie de Fibonacci recursiva.

```
public static int fib (int n) {  
    if (n==0 || n==1)  
        return 1;  
    else  
        return fib(n-1) + fib(n-2);  
}
```

El profesor realizará una demostración del funcionamiento de SRec en la que mostrará, mediante este algoritmo recursivo múltiple, el funcionamiento de sus funciones principales: seleccionar una máquina virtual de Java, procesar un fichero, generar y reproducir una animación, manejar paneles, almacenar y cargar la animación, y cambiar las diversas opciones de configuración.

2. **Primera animación.** Duración aproximada: 15 minutos. Algoritmo: exponenciación mediante partición binaria del exponente.

```
public static int pot (int b, int e) {
    if (e==0)
        return 1;
    else if (e%2==0)
        return pot(b*b,e/2);
    else
        return b*pot(b*b,e/2);
}
```

El alumno generará una animación de este algoritmo recursivo lineal. Para ello, debe realizar las siguientes tareas:

- i. Procesar la clase *ClaseEvaluacion* disponible en el sitio *web* de la asignatura (menú de Archivo).
 - ii. Generar una animación (menú de Archivo).
 - iii. Experimentar con todos los controles de animación hasta que se comprenda su funcionamiento.
 - iv. Almacenar la animación y cargarla de nuevo (ambas operaciones, en menú de Archivo).
3. **Segunda animación.** Duración aproximada: 20 minutos. Algoritmo: números combinatorios.

```
public static int comb (int m, int n) {
    if (n==0)
        return 1;
    else if (m==n)
        return 1;
    else
        return comb(m-1,n) + comb(m-1,n-1);
}
```

El alumno experimentará con una animación de este algoritmo recursivo múltiple. Esta vez se le pide que realice las siguientes tareas:

- v. Generar una animación.
- vi. Experimentar con los controles de animación (parte superior derecha de la ventana), observando el efecto de cada operación sobre cada una de las vistas de la recursividad:
 1. Traza.
 2. Pila de control.
 3. Árbol de activación.
- vii. Experimentar con las opciones de configuración (menú Configuración) de:
 1. Control de la cantidad de información mostrada en cada nodo del árbol.
 2. Control de la visualización de la historia pasada.

3. Control del formato gráfico de las 3 visualizaciones (traza, pila de control, árbol de activación).

4. **Tercera animación.** Duración aproximada: 40 minutos. Problema: competición.

Dos participantes A y B juegan una competición que es ganada por el primero que venza en n partidos, $n > 0$. En principio, ambos participantes tienen cualidades y preparación similares, por lo que cada uno tiene un 50% de probabilidad de ganar cada partido. Se quiere conocer la probabilidad que tiene el equipo A de ganar la competición si se sabe que A necesita i partidos para ganar y B necesita j . Obsérvese que al menos $i > 0$ ó $j > 0$ para que la situación tenga sentido.

Podemos denotar el problema como $prob(i,j)$. Una solución directa es la siguiente:

```
public static float prob (int i, int j) {
    if (i==0)
        return (float)1.0;
    else if (j==0)
        return (float)0.0;
    else
        return (float)((prob(i, j-1)+prob(i-
1, j))/2.0);
}
```

Se pretende eliminar la redundancia existente en este algoritmo recursivo múltiple. Durante el resto de la sesión se pide desarrollar un árbol de recursión y su grafo de activación asociado, que sean representativos del algoritmo. Puede servirse de SRec o estudiar el código de forma estática.

Entrega

Ambas representaciones gráficas se entregarán al profesor de la asignatura en una hoja (con los apellidos del alumno).

Posteriormente, estas representaciones se usarán como base para aplicar las técnicas de memorización y de tabulación. El plazo de entrega de la práctica completa es el 10 de diciembre de 2007. Debe enviarse por correo electrónico, adjuntando un breve informe siguiendo el modelo disponible en el sitio *web* de la asignatura.

5. **Cuestionario.** Duración aproximada: 20 minutos.
El alumno debe responder a las preguntas siguientes.

Cuestionario de opinión sobre el sistema SRec

Nombre y apellidos (opcional): _____

En las preguntas siguientes, marca un valor en cada pregunta. Debes usar un valor de la escala mostrada en la siguiente tabla. Según la clase de pregunta, su significado se referirá a opinión o calidad:

Valor	Opinión	Calidad
1	Nada de acuerdo	Muy mala
2	Poco de acuerdo	Mala
3	Sin opinión	Regular
4	Algo de acuerdo	Buena
5	Totalmente de acuerdo	Muy buena

Te parece que SRec es **fácil de usar**: []

Las partes que te parecen más **difíciles de usar** (si las hay) son:

Te parece que SRec **te ha ayudado** a analizar los algoritmos recursivos para:

- [] Analizar qué llamadas se realizan en tiempo de ejecución
- [] Identificar las dependencias entre llamadas

Te parece que, **la calidad en general** de SRec para analizar la recursividad es alta: []

Las partes de **mejor calidad**, para ti, son:

Las partes de **peor calidad**, para ti, son:

Te parece que **la calidad de varias partes** de SRec es alta:

- [] Estructura del menú principal
- [] Iconos
- [] Controles de animación
- [] Vista de traza
- [] Vista de pila de control
- [] Vista de árbol de activación
- [] Configuración de las visualizaciones
- [] Interacción con los paneles (*scroll*, mover paneles, *zoom*)

En conjunto, **te ha gustado** SRec: []

Responde a las siguientes preguntas en formato libre:

1. Di qué características te parece que podrían ser **útiles** pero SRec carece de ellas:

2. Di qué características de SRec te parecen tan **poco útiles** que las suprimirías:

3. Describe los **aspectos positivos** que encuentras en SRec (sobre todo si no se han mencionado antes):

4. Describe los **aspectos negativos** que encuentras en SRec (sobre todo si no se han mencionado antes)

C Enunciado y cuestionario de la Sesión 3

Ingeniería Informática Asignatura *Diseño y Análisis de Algoritmos*

Curso 2008/2009 Práctica nº 2

Objetivo

El objetivo de la práctica es que el alumno practique en la eliminación de la recursividad múltiple.

Carácter

La práctica es voluntaria. La sesión en el laboratorio se realizará individualmente. El resto de la práctica puede realizarse en parejas.

Prerrequisitos

El alumno debe tener nociones básicas de eliminación de la recursividad múltiple.

Enunciado

SRec es una aplicación para la visualización y animación de métodos recursivos. La práctica consta de dos partes, una a realizar en el laboratorio y otra, fuera del mismo.

La sesión de laboratorio también sirve para evaluar la calidad de SRec. La sesión consta de cuatro fases a realizar secuencialmente:

1. **Demostración del profesor.** Duración aproximada: 15 minutos. Algoritmo: serie de Fibonacci recursiva.

```
public static int fib (int n) {  
    if (n==0 || n==1)  
        return 1;  
    else  
        return fib(n-1) + fib(n-2);  
}
```

El profesor realizará una demostración del funcionamiento de SRec en la que mostrará, mediante este algoritmo recursivo múltiple, el funcionamiento de sus funciones principales: seleccionar una máquina virtual de Java, procesar un

fichero, generar y reproducir una animación, manejar paneles, almacenar y cargar la animación, y cambiar las diversas opciones de configuración.

2. **Primera animación.** Duración aproximada: 15 minutos. Algoritmo: exponenciación mediante partición binaria del exponente.

```
public static int pot (int b, int e) {
    if (e==0)
        return 1;
    else if (e%2==0)
        return pot(b*b,e/2);
    else
        return b*pot(b*b,e/2);
}
```

El alumno generará una animación de este algoritmo recursivo lineal. Para ello, debe realizar las siguientes tareas:

- viii. Procesar la clase *ClaseEvaluacion* disponible en el sitio *web* de la asignatura (menú de Archivo).
- ix. Generar una animación (menú de Archivo).
- x. Experimentar con todos los controles de animación hasta que se comprenda su funcionamiento.
- xi. Almacenar la animación y cargarla de nuevo (ambas operaciones, en menú de Archivo).

3. **Segunda animación.** Duración aproximada: 20 minutos. Algoritmo: números combinatorios.

```
public static int comb (int m, int n) {
    if (n==0)
        return 1;
    else if (m==n)
        return 1;
    else
        return comb(m-1,n) + comb(m-1,n-1);
}
```

El alumno experimentará con una animación de este algoritmo recursivo múltiple. Esta vez se le pide que realice las siguientes tareas:

- xii. Generar una animación.
- xiii. Experimentar con los controles de animación (parte superior derecha de la ventana), observando el efecto de cada operación sobre cada una de las vistas de la recursividad:
 1. Traza.
 2. Pila de control.
 3. Árbol de activación.

- xiv. Experimentar con las opciones de configuración (menú Configuración) de:
1. Control de la cantidad de información mostrada en cada nodo del árbol.
 2. Control de la visualización de la historia pasada.
 3. Control del formato gráfico de las 3 visualizaciones (traza, pila de control, árbol de activación).

4. **Tercera animación.** Duración aproximada: 40 minutos. Algoritmo: competición.

Dos participantes A y B juegan una competición que es ganada por el primero que venza en n partidos, $n > 0$. En principio, ambos participantes tienen cualidades y preparación similares, por lo que cada uno tiene un 50% de probabilidad de ganar cada partido. Se quiere conocer la probabilidad que tiene el equipo A de ganar la competición si se sabe que A necesita i partidos para ganar y B necesita j . Obsérvese que al menos $i > 0$ ó $j > 0$ para que la situación tenga sentido.

Podemos denotar el problema como $prob(i, j)$. Una solución directa es la siguiente:

```
public static float prob (int i, int j) {
    if (i==0)
        return (float)1.0;
    else if (j==0)
        return (float)0.0;
    else
        return(float)((prob(i, j-1)+prob(i-1, j))/2.0);
}
```

Se pretende eliminar la redundancia existente en este algoritmo recursivo múltiple. Durante el resto de la sesión se pide desarrollar un árbol de recursión y su grafo de activación asociado, que sean representativos del algoritmo. Puede servirse de SRec o estudiar el código de forma estática.

Entrega

Ambas representaciones gráficas se entregarán al profesor de la asignatura en una hoja (con los apellidos del alumno).

Posteriormente, estas representaciones se usarán como base para aplicar las técnicas de memorización y de tabulación. El plazo de entrega de la práctica completa es el 20 de noviembre de 2007. Debe enviarse por correo electrónico, adjuntando un breve informe siguiendo el modelo disponible en el sitio *web* de la asignatura. El mensaje debe enviarse a Antonio Pérez Carrasco (antonio.perez.carrasco@urjc.es) con copia a Ángel Velázquez (angel.velazquez@urjc.es).

5. **Cuestionario.** Duración aproximada: 20 minutos.
El alumno debe responder a las preguntas del cuestionario que se entregará en papel.

Cuestionario

Nombre y apellidos (opcional): _____

En las preguntas siguientes, marca un valor en cada pregunta. Debes usar un valor de la escala mostrada en la siguiente tabla. Según la clase de pregunta, su significado se referirá a opinión o calidad.

Valor	Opinión	Calidad
1	Nada de acuerdo	Muy mala
2	Poco de acuerdo	Mala
3	Sin opinión	Regular
4	Algo de acuerdo	Buena
5	Totalmente de acuerdo	Muy buena

Si te parece que SRec es fácil de usar

Las partes que te parecen más difíciles de usar (si las hay) son:

Si te parece que SRec te ha ayudado a analizar los algoritmos recursivos para:

Determinar las llamadas recursivas que se realizan en tiempo de ejecución

Determinar las dependencias entre llamadas recursivas

Si te parece alta la calidad en general de SRec para analizar la recursividad

Si te parece alta la calidad de varios aspectos de SRec:

Iconos

Controles de animación

Vista de árbol de activación

Visor de árboles grandes

Configuración de formatos

Configuración de zoom

Interacción con los paneles (*scroll*, mover paneles, mostrar/ocultar paneles)

Proceso de generación de una animación

Proceso de almacenar/cargar una animación

Visualización almacenada en un fichero de captura

Si en conjunto te ha gustado SRec

Responde a las siguientes preguntas en formato libre:

1. Di qué características te parece que podrían ser útiles pero SRec carece de ellas:

2. Di qué características de SRec te parecen tan poco útiles que las suprimirías:

3. Describe los aspectos positivos que encuentras en SRec (sobre todo si no se han mencionado antes)

4. Describe los aspectos negativos que encuentras en SRec (sobre todo si no se han mencionado antes)

D Respuestas íntegras de la Sesión 1

Tabla 5. Respuestas íntegras de los alumnos participantes

SRec me parece fácil de usar	4	4	5	4
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para buscar el error	2	2	3	3
SRec me ha ayudado a analizar algoritmos recursivos para comprobar que la solución propuesta es la correcta	5	4	5	5
Evalúa la calidad en general de SRec para analizar la recursividad	4	5	-	5
Las partes de mejor calidad, para ti, son:	La traza del programa y el manejo de las animaciones	Hallar soluciones a algoritmos recursivos introducidos	La animación del árbol como la de la pila	La visualización del árbol
Las partes de peor calidad, para ti, son:	-	Encontrar errores en algoritmos introducidos	La parte de código, ya que no se puede modificar	No seguimiento del código
Evalúa la calidad de las siguientes partes de SRec:				
Estructura del menú principal	5	4	4	5
Controles de animación	5	5	5	5
Vista de traza	4	5	4	5
Vista de pila de control	4	5	4	5
Vista de árbol de activación	5	4	5	5
Configuración de las visualizaciones	4	5	5	4
Interacción con los paneles (Scroll, mover paneles, zoom)	5	4	4	4
En conjunto, SRec me ha gustado	4	4	4	4

Di qué características te parece que podrían ser útiles y SRec carece de ellas	Una característica interesante sería poder ver los pasos que está dando el programa, es decir, resaltar la instrucción que se está ejecutando	Un compilador que muestra errores de sintaxis en el código. Más opciones de desarrollo y edición de las animaciones	En la ventana de código se podría mostrar o ir señalando qué instrucciones se están ejecutando, y dar la opción de ver valores de variables del código	Un seguimiento de cada línea que se va ejecutando en el código
Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	Creo que todas son útiles	Ninguna	-
Describe los aspectos positivos que encuentras en SRec	Su fácil manejo y gran ayuda en vislumbrar la recursión de los programas	Sencillez. Animaciones claras. Educativo y útil para el desarrollo de ejercicios	La personalización y animaciones. Facilidad de manejo.	Simplicidad
Describe los aspectos negativos que encuentras en SRec	-	No ayuda de forma práctica a la obtención de los errores	La carga de archivo java cada vez que se modifica, un botón recargar agilizaría la búsqueda	Pocas herramientas y tener que procesar el código cada vez que actualizamos el fichero

SRec me parece fácil de usar	5	4	5
Las partes que te parecen más difíciles de usar (si las hay) son:	-	Desplazamiento por árboles con muchos nodos. Dificil visión global del árbol	Ninguna
SRec me ha ayudado a analizar algoritmos recursivos para buscar el error	5	2	4
SRec me ha ayudado a analizar algoritmos recursivos para comprobar que la solución propuesta es la correcta	5	5	4
Evalúa la calidad en general de SRec para analizar la recursividad	5	3	5
Las partes de mejor calidad, para ti, son:	Control total sobre la ejecución y tratamiento del algoritmo	Manejo de las llamadas con los distintos controles: paso a paso, atrás, play, etc.	Árbol de recursión

Las partes de peor calidad, para ti, son:	Quizás que la traza fuese más completa, mostrando variables que entran en juego en el algoritmo a tratar	No poder hacer traza de código. Tener que hacer mucho zoom para ver todos los nodos, aunque los juntes	Ninguna
Evalúa la calidad de las siguientes partes de SRec:			
Estructura del menú principal	4	4	4
Controles de animación	5	5	5
Vista de traza	5	3	4
Vista de pila de control	5	4	5
Vista de árbol de activación	5	5	5
Configuración de las visualizaciones	5	4	4
Interacción con los paneles (Scroll, mover paneles, zoom)	5	3	4
En conjunto, SRec me ha gustado	5	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Como dije antes, más opciones de traza	Traza de código. Poner atajos de teclado para los controles de la animación	Resaltar aquellos nodos del árbol que han producido un error de diferente color a aquellos que han ¿? la ejecución
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Para al menos estos ejemplos, la pila no tiene gran relevancia	Tipo y grosor de flechas	-
Describe los aspectos positivos que encuentras en SRec	Control total del árbol de recursividad	Intuitiva visión de las llamadas recursivas: rama activa, pila, nodos ya tratados...	Toda la configuración de tamaño, letras, tipos, etc. Y muy interesante la opción de zoom
Describe los aspectos negativos que encuentras en SRec	Lo dicho antes sobre las opciones de traza	Usar el atajo Ctrl+Z para configurar el zoom cuando suele ser "retroceder"	-

E Respuestas íntegras de la Sesión 2

Tabla 6. Respuestas íntegras de los alumnos participantes

SRec me parece fácil de usar	5	4	4
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	5	4	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	5	4	3
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	5	4	4
Las partes de mejor calidad, para ti, son:	La pila de control y el árbol de activación	Las animaciones	Verificar el coste en memoria mediante la pila de ejecución
Las partes de peor calidad, para ti, son:	La vista de la traza	-	-
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	5	3	4
Iconos	5	2	3
Controles de animación	5	4	4
Vista de traza	5	4	4
Vista de pila de control	5	4	3
Vista de árbol de activación	5	4	4
Configuración de las visualizaciones	5	3	3
Interacción con los paneles (Scroll, mover paneles, zoom)	5	3	2
En conjunto, SRec me ha gustado	-	4	4

Di qué características te parece que podrían ser útiles y SRec carece de ellas	El cálculo de la complejidad en tiempo y en memoria de un algoritmo	No se puede imprimir. Que Ctrl+Scroll modifique el zoom de la animación	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Todo me parece muy útil	-	-
Describe los aspectos positivos que encuentras en SRec	La posibilidad de ver poco a poco las llamadas recursivas	-	-
Describe los aspectos negativos que encuentras en SRec	Ninguna	-	-

SRec me parece fácil de usar	4	5	4
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	4	5	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	4	4	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	4
Las partes de mejor calidad, para ti, son:	Animaciones	Interfaz sencillo	Interfaz sencilla
Las partes de peor calidad, para ti, son:	Barra de herramientas	Adaptaciones de árboles grandes	Ajuste del árbol y cierre de paneles
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	3	5	4
Iconos	3	4	3
Controles de animación	5	4	4
Vista de traza	5	4	4
Vista de pila de control	4	4	3
Vista de árbol de activación	4	4	3
Configuración de las visualizaciones	4	3	4

Interacción con los paneles (Scroll, mover paneles, zoom)	4	3	3
En conjunto, SRec me ha gustado	-	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Una traza sobre el código visible y la generación de un diagrama de flujo	-	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Ninguna, todo puede ser útil	-	-
Describe los aspectos positivos que encuentras en SRec	Poder visualizar la traza personalizándolo como se quiera	-	Es una interfaz sencilla y útil para hacer trazos en nuestro programas Java
Describe los aspectos negativos que encuentras en SRec	-	-	Quizá sea fallo de manejo mía pero me parece que los ajustes del árbol no son muy buenos ya que el botón no funciona del todo bien

SRec me parece fácil de usar	5	5	4
Las partes que te parecen más difíciles de usar (si las hay) son:	Ajustar el zoom con árboles grandes en pantalla completa	-	-
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	5	4	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	5	4	5
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	5	4	4
Las partes de mejor calidad, para ti, son:	Las 4 pantallas, sobre todo la parte donde te van mostrando las variables (parte inferior izquierda)	El árbol de recursión	Animación, pila, árbol
Las partes de peor calidad, para ti, son:	-	La traza	Traza

Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	5	3	4
Iconos	5	3	4
Controles de animación	5	4	5
Vista de traza	4	3	3
Vista de pila de control	5	4	5
Vista de árbol de activación	5	5	5
Configuración de las visualizaciones	4	3	4
Interacción con los paneles (Scroll, mover paneles, zoom)	4	4	4
En conjunto, SRec me ha gustado	5	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	A lo mejor que te vaya poniendo en el árbol la operación que vas realizando, ya que son árboles	Poder configurar el tamaño de las flechas solo, así se podrá ver gandes las llamadas y más juntas o separadas	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	La visualización del código, ya que quita espacio para poder ver más grande el árbol	La traza no parece que ¿redunde? Ya que con la pila se ve muy bien	Guardar las animaciones ya que el proceso de crearlas es rápido. Aunque también es útil para trabajos más grandes
Describe los aspectos positivos que encuentras en SRec	Todas las configuraciones de colores y posibilidad de dar formato al árbol para así comprender mejor el algoritmo	-	El ir creando el árbol de recursión paso a paso y así ver la pila con detalle
Describe los aspectos negativos que encuentras en SRec	-	-	-

SRec me parece fácil de usar	4	5	5
Las partes que te parecen más difíciles de usar (si las hay) son:			
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	4	4	4

SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	5	4	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	4
Las partes de mejor calidad, para ti, son:	La posibilidad de ver las diferentes partes de la recursividad: árbol, pila y la traza, junto con el código	-	Animaciones
Las partes de peor calidad, para ti, son:	La parte visual, en el caso de llamadas numerosas, se queda algo pequeña	La exportación de la animación a un GIF	Exportar animaciones como GIF
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	3	4	4
Iconos	4	4	4
Controles de animación	4	4	4
Vista de traza	4	3	3
Vista de pila de control	4	4	3
Vista de árbol de activación	3	5	4
Configuración de las visualizaciones	2	5	3
Interacción con los paneles (Scroll, mover paneles, zoom)	3	5	4
En conjunto, SRec me ha gustado	4	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Una pequeña ventana con una vista "total" del árbol que en el caso de un zoom amplio nos permitiría mover fácilmente a otra opción del árbol (como un "minimap")	-	Otro tipo de animaciones, como grafos de dependencia, etc.
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Hay demasiadas opciones para configurar el aspecto gráfico. Agradecería la existencia de algunos temas	-	-
Describe los aspectos positivos que encuentras en SRec	-	-	Claridad a la hora de probar un código para ver lo que hace

Describe los aspectos negativos que encuentras en SRec	-	La exportación de la imagen GIF implica tener abierto el programa y estar visionándolo y no permite la ejecución de otras tareas a la vez	Exportar animaciones como GIF (lentitud y no poder hacer nada mientras se exporta).
--	---	---	---

SRec me parece fácil de usar	4	5	5
Las partes que te parecen más difíciles de usar (si las hay) son:			
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	3	5	5
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	4	5	5
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	5
Las partes de mejor calidad, para ti, son:	-	-	Visualización de las llamadas y la dependencia entre ellas
Las partes de peor calidad, para ti, son:	-	-	No he tenido tiempo para descubrirlas
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	3	5	5
Iconos	3	5	4
Controles de animación	4	5	5
Vista de traza	3	5	5
Vista de pila de control	3	5	5
Vista de árbol de activación	4	5	5
Configuración de las visualizaciones	4	5	4
Interacción con los paneles (Scroll, mover paneles, zoom)	4	4	5
En conjunto, SRec me ha gustado	4	4	5

Di qué características te parece que podrían ser útiles y SRec carece de ellas	Realizar el grafo de dependencia	Al ejecutar he tenido un error porque el programa ha lanzado una excepción. Creo que sería útil decir qué tipo de excepción se ha producido.	Utilización de algún otro lenguaje como C, además de Java
Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	La opción de animación hacia atrás	No creo que sea imprescindible poder modificar los colores del árbol
Describe los aspectos positivos que encuentras en SRec	Útil para comprender la recursión	Me parece una herramienta muy útil por todas las opciones que ofrece, además de ser sencilla	Claridad en la visualización de la traza y el árbol de derivación
Describe los aspectos negativos que encuentras en SRec	-	Lo mencionado en la pregunta 1ª	Ninguno por el momento

SRec me parece fácil de usar	5	4	5
Las partes que te parecen más difíciles de usar (si las hay) son:			
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	4	5	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	4	4	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	5	5
Las partes de mejor calidad, para ti, son:	Animación	La parte gráfica, es decir, la representación	Interfaz, visualización
Las partes de peor calidad, para ti, son:	La ventana de llamadas recursivas, muy poco visual	-	-
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	4	4	4
Iconos	5	4	4
Controles de animación	5	5	4

Vista de traza	3	4	5
Vista de pila de control	4	4	5
Vista de árbol de activación	5	5	5
Configuración de las visualizaciones	5	5	4
Interacción con los paneles (Scroll, mover paneles, zoom)	4	5	4
En conjunto, SRec me ha gustado	4	4	5
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Indicar qué instrucciones se van ejecutando, con la animación. Una ventana con el valor de las variables durante la ejecución para depurar el código	Indicar el tiempo de ejecución, la parte del código concreto que se está ejecutando.	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	-	-
Describe los aspectos positivos que encuentras en SRec	La animación facilita entender	Su gran potencia para estudiar la recursividad	Permite visualizar la ejecución paso a paso de algoritmos recursivos con lo que es fácil la traza del algoritmo
Describe los aspectos negativos que encuentras en SRec	-	-	-

SRec me parece fácil de usar	5	4	4
Las partes que te parecen más difíciles de usar (si las hay) son:	-	Cargar clases, generar animación, modificarla	La selección de la función que se quiere animar
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	4	4	5
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	5	4	5
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	5

Las partes de mejor calidad, para ti, son:	Interfaz intuitivo	Animación	Los botones de control de la animación
Las partes de peor calidad, para ti, son:	-	Problemas con la recuperación de archivos, no encuentro clases ni el path es excesivamente largo	-
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	5	4	4
Iconos	5	3	5
Controles de animación	4	3	5
Vista de traza	5	4	4
Vista de pila de control	5	4	4
Vista de árbol de activación	5	3	5
Configuración de las visualizaciones	4	3	3
Interacción con los paneles (Scroll, mover paneles, zoom)	4	3	5
En conjunto, SRec me ha gustado	5	-	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	-	Independencia de las ventanas internas, o sea, que se puedan extraer, tipo Photoshop, así se aumentaría el espacio de trabajo. Un minimap, el grafo es grande para saber en qué posición estamos	Exportar los árboles a formatos como JPG, PDF, etc.
Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	El poder cambiar los colores. Se podrían poner más por defecto ya que una vez que se aprenden, el cambio podría generar confusión	-
Describe los aspectos positivos que encuentras en SRec	Muestra y ocultación de la parte a usar (código, árbol)	-	-
Describe los aspectos negativos que encuentras en SRec	-	-	-
SRec me parece fácil de usar	5	5	4

Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	5	4	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	5	5	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	5	5	4
Las partes de mejor calidad, para ti, son:	La animación de árboles de recursión (en general, la interfaz)	Aniamción, traza, pila de control	Generar el árbol de recursividad
Las partes de peor calidad, para ti, son:	-	-	Imagen GIF
Te parece que la calidad de varias partes de SRec es alta:			
Estructura del menú principal	5	4	3
Iconos	4	5	3
Controles de animación	5	5	4
Vista de traza	5	4	5
Vista de pila de control	3	5	3
Vista de árbol de activación	5	4	3
Configuración de las visualizaciones	4	3	4
Interacción con los paneles (Scroll, mover paneles, zoom)	4	4	4
En conjunto, SRec me ha gustado	5	5	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Podría ser útil la generación del grafo de dependencia además del árbol de recursión	Ajuste de zoom automático según va creciendo el árbol y/o desplazamiento automático hacia donde está ejecutando	Cambia el generar GIF, ya que falla. Se puede generar una imagen JPG. Se puede acceder desde cualquier ruta
Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	-	-

Describe los aspectos positivos que encuentras en SRec	La interfaz gráfica que te permite ver tanto el árbol, código, traza y pila de control de forma que puedas seguir correctamente toda la ejecución de un algoritmo.	La calidad gráfica (colores, sombreado...) y la magnífica ordenación	Ver la dependencia optimizada
Describe los aspectos negativos que encuentras en SRec	Imposibilidad de colocar los paneles en otra posición (por ejemplo colocarlos verticalmente uno sobre otro). Sólo he conseguido aumentar y disminuir su tamaño en la misma posición	Ninguno	-

SRec me parece fácil de usar	4	5	4	4
Las partes que te parecen más difíciles de usar (si las hay) son:	Demasiadas configuraciones	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para analizar qué llamadas se realizan en tiempo de ejecución	5	4	3	5
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas	5	4	4	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	4	4
Las partes de mejor calidad, para ti, son:	El árbol de recursión	Las presentaciones de gráficos	La pila y el árbol de recursión	Árbol de recursión
Las partes de peor calidad, para ti, son:	Guardar imagen GIF (si cambias la pantalla se copia)	A veces tarda mucho para grandes	La traza	No hay partes de baja calidad, pero todo es mejorable
Te parece que la calidad de varias partes de SRec es alta:				
Estructura del menú principal	4	5	4	4
Iconos	3	5	3	3
Controles de animación	5	5	5	5
Vista de traza	4	4	3	3

Vista de pila de control	4	3	4	4
Vista de árbol de activación	5	5	5	4
Configuración de las visualizaciones	4	4	5	3
Interacción con los paneles (Scroll, mover paneles, zoom)	3	5	4	3
En conjunto, SRec me ha gustado	4	-	-	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	No se me ocurre ninguna, pero mejoraría lo de crear el GIF	-	Admitir otros lenguajes aparte de Java. Comprobar el estado de las variables para poder ir estudiando el código	Permitir cargar código de otros lenguajes como por ejemplo C o C++
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Ninguna	-	-	La generación de las llamadas, ya se observa en el árbol
Describe los aspectos positivos que encuentras en SRec	Lo mejor es ver la dependencia y así poder optimizar los algoritmos	Buenos gráficos, facilidad de uso, interfaz amigable e intuitiva	Contiene un compilador para corregir errores del código. Es flexible y fácil de usar	Se ve muy rápido el proceso de recursión.
Describe los aspectos negativos que encuentras en SRec	Errores al cargar archivos de ciertas rutas	Para números grandes tarda mucho. Quizá se pudiera mejorar la forma de mostrar errores	-	-

F Respuestas íntegras de la Sesión 3

Tabla 7. Respuestas íntegras de los alumnos participantes

SRec me parece fácil de usar	4	4	5
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	4	5	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	4	4	3
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	5
Te parece que la calidad de varias partes de SRec es alta:			
Iconos	4	4	4
Controles de animación	4	5	3
Vista de árbol de activación	4	4	5
Visor de árboles grandes	4	3	5
Configuración de formatos	4	3	4
Configuración del zoom	4	4	5
Interacción con los paneles (Scroll, mover paneles, zoom)	4	5	3
Proceso de generación de una animación	4	4	4
Proceso de almacenar/cargar una animación	4	4	4
Visualización almacenada de un fichero de captura	4	5	4
En conjunto, SRec me ha gustado	4	4	4

Di qué características te parece que podrían ser útiles y SRec carece de ellas	Ninguna	-	Una herramienta que dibuje los grafos de dependencia
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Cambio de colores en los nodos	-	-
Describe los aspectos positivos que encuentras en SRec	Ninguno	-	-
Describe los aspectos negativos que encuentras en SRec	Cuando los valores de entrada son altos, no se ve bien la animación	-	-

SRec me parece fácil de usar	-	5	4
Las partes que te parecen más difíciles de usar (si las hay) son:			
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	4	4	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	4	5	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	5	4
Te parece que la calidad de varias partes de SRec es alta:			
Iconos	4	4	2
Controles de animación	4	4	3
Vista de árbol de activación	4	5	4
Visor de árboles grandes	4	5	3
Configuración de formatos	4	5	3
Configuración del zoom	4	2	3
Interacción con los paneles (Scroll, mover paneles, zoom)	4	4	3
Proceso de generación de una animación	4	4	4

Proceso de almacenar/cargar una animación	4	4	4
Visualización almacenada de un fichero de captura	4	3	4
En conjunto, SRec me ha gustado	4	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	-	Exportar a JPEG, a fin de ahorrar capturas de pantalla en la memoria	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Lo de cambiar de color	Ninguna	-
Describe los aspectos positivos que encuentras en SRec	-	-	Facilidad de seguir los pasos de algoritmos recursivos
Describe los aspectos negativos que encuentras en SRec	-	La manera de exportar a GIF, sin que se pueda ajustar el zoom "in situ". No es malo, pero el primer par de veces cuesta cogerle la mecánica	-

SRec me parece fácil de usar	4	4	5
Las partes que te parecen más difíciles de usar (si las hay) son:	No se puede interrumpir la creación de un GIF	-	-
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	5	4	3
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	5	4	3
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	5	3	3
Te parece que la calidad de varias partes de SRec es alta:			
Iconos	4	2	4
Controles de animación	2	3	4

Vista de árbol de activación	4	4	5
Visor de árboles grandes	5	3	4
Configuración de formatos	5	3	3
Configuración del zoom	5	4	3
Interacción con los paneles (Scroll, mover paneles, zoom)	5	4	3
Proceso de generación de una animación	5	4	4
Proceso de almacenar/cargar una animación	5	3	4
Visualización almacenada de un fichero de captura	4	4	3
En conjunto, SRec me ha gustado	5	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Presentar una captura de gran tamaño (poster) para árboles grandes	Aparte del árbol podría ofrecer el grafo en el que se muestran realmente las dependencias de forma más directa	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Ninguna	Hay iconos que me parecen poco intuitivos, más que suprimirlos, los cambiaría	-
Describe los aspectos positivos que encuentras en SRec	-	Nos ofrece una clara visión del árbol y por tanto, de la recursividad, aparte de ofrecer la salida del algoritmo de forma detallada	-
Describe los aspectos negativos que encuentras en SRec	-	La visión del zoom en caso de árboles muy grandes no permite ver el árbol de forma óptima	-
SRec me parece fácil de usar	4	3	4
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	5	4	5

SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	5	2	5
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	3	4
Te parece que la calidad de varias partes de SRec es alta:			
Iconos	5	3	3
Controles de animación	4	3	4
Vista de árbol de activación	4	3	4
Visor de árboles grandes	4	4	3
Configuración de formatos	4	2	4
Configuración del zoom	3	2	4
Interacción con los paneles (Scroll, mover paneles, zoom)	5	2	4
Proceso de generación de una animación	3	3	5
Proceso de almacenar/cargar una animación	5	3	5
Visualización almacenada de un fichero de captura	5	3	5
En conjunto, SRec me ha gustado	4	3	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Menú "Ver" con distintas opciones, con pantalla completa, más zoom, menos zoom, ...	-	-
Di qué características de SRec te parecen tan poco útiles que las suprimirías	El formato tipográfico, dado que la aplicación tiene carácter didáctico y no sirve de mucho cambiar el color para ello	-	-
Describe los aspectos positivos que encuentras en SRec	Lo ilustrativo que es en la muestra de la recursividad	Que se puede seguir el algoritmo paso a paso	-
Describe los aspectos negativos que encuentras en SRec	Configuración con más libertad de la barra de herramientas	En general, la dificultad de la instalación	-
SRec me parece fácil de usar	4	4	4

Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	4	4	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	4	3	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	3	4	3
Te parece que la calidad de varias partes de SRec es alta:			
Iconos	4	2	3
Controles de animación	4	4	3
Vista de árbol de activación	3	4	3
Visor de árboles grandes	3	2	4
Configuración de formatos	4	3	4
Configuración del zoom	1	4	4
Interacción con los paneles (Scroll, mover paneles, zoom)	3	4	3
Proceso de generación de una animación	2	3	4
Proceso de almacenar/cargar una animación	4	3	5
Visualización almacenada de un fichero de captura	4	4	4
En conjunto, SRec me ha gustado	4	3	3
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Un ejecutable normal y corriente	Que realice también el grafo de dependencia	No es multiplataforma
Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	Los iconos no me parecen los más adecuados	La forma de cargar archivos
Describe los aspectos positivos que encuentras en SRec	-	Es una buena ayuda para el estudio de la recursividad	Implementación fácil

Describe los aspectos negativos que encuentras en SRec	Muy lento con algunas animaciones grandes	El icono de "carga de una clase"	Limitación en su implementación
SRec me parece fácil de usar	4	4	5
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	-
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	5	X	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	4	-	5
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	5	X	4
Te parece que la calidad de varias partes de SRec es alta:			
Iconos	5	-	3
Controles de animación	4	X	3
Vista de árbol de activación	4	X	4
Visor de árboles grandes	5	-	4
Configuración de formatos	3	-	4
Configuración del zoom	5	-	4
Interacción con los paneles (Scroll, mover paneles, zoom)	-	-	4
Proceso de generación de una animación	5	X	5
Proceso de almacenar/cargar una animación	4	-	5
Visualización almacenada de un fichero de captura	4	-	5
En conjunto, SRec me ha gustado	5	X	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Crear grafos de dependencia	-	El programa está muy bien diseñado

Di qué características de SRec te parecen tan poco útiles que las suprimirías	-	El formato tipográfico de la función, que tiene demasiadas opciones	Todavía no me he encontrado con ninguna
Describe los aspectos positivos que encuentras en SRec	Zoom, exportar animación a GIF	Rápida compilación y generación de animaciones	Fácil manejo
Describe los aspectos negativos que encuentras en SRec	-	Demasiados aspectos en la interfaz, al principio es un poco desconcertante	Ninguno hasta ahora

SRec me parece fácil de usar	5	4	4	4
Las partes que te parecen más difíciles de usar (si las hay) son:	-	-	Configuraciones acerca de archivos intermedios, etc.	-
SRec me ha ayudado a analizar algoritmos recursivos para determinar las llamadas recursivas que se realizan en tiempo de ejecución	4	4	4	4
SRec me ha ayudado a analizar algoritmos recursivos para identificar las dependencias entre llamadas recursivas	5	4	4	4
Te parece que, la calidad en general de SRec para analizar la recursividad es alta	4	4	5	4
Te parece que la calidad de varias partes de SRec es alta:				
Iconos	4	4	4	3
Controles de animación	5	4	4	4
Vista de árbol de activación	3	4	4	5
Visor de árboles grandes	5	4	3	4
Configuración de formatos	4	4	5	4
Configuración del zoom	4	4	4	5
Interacción con los paneles (Scroll, mover paneles, zoom)	5	4	3	4
Proceso de generación de una animación	5	4	4	4
Proceso de almacenar/cargar una animación	5	4	4	4

Visualización almacenada de un fichero de captura	5	4	4	-
En conjunto, SRec me ha gustado	4	4	4	4
Di qué características te parece que podrían ser útiles y SRec carece de ellas	Dibujo del grafo asociado al árbol	No sé si puedes modificar el algoritmo en una pantalla de edición	Aún no sé ninguna	Grafo de dependencia
Di qué características de SRec te parecen tan poco útiles que las suprimirías	Cambio en apariencia de todo el conjunto	No he encontrado ninguna	Ninguna	-
Describe los aspectos positivos que encuentras en SRec	La ayuda de la traza. Guarda en formato XML, muy estándar	El poder generar árboles de recursión grandes a través de la animación	Muy instructivo y visual	-
Describe los aspectos negativos que encuentras en SRec	-	Hay veces que el zoom no logra ver todo	Manejar árboles grandes, guardar animación (si es grande)	-