

SSL 2005.1 – Presentación General del Trabajo Práctico – Simulación de un Año Académico de SSL.

1.0.0.20050501

por José María Sola y Jorge Muchnik.

Este documento es una presentación general. Su contenido detallado puede sufrir modificaciones a lo largo del desarrollo del trabajo práctico, siempre en beneficio de una mayor claridad para el desarrollo del mismo.

Contenidos

SSL 2005.1 – Presentación General del Trabajo Práctico – Simulación de un Año Académico de SSL.....	1
Contenidos.....	1
Objetivos	1
Medios para Cumplir estos Objetivos	1
Contexto del Trabajo Práctico -- Simulación de un Año Académico de SSL.....	1
Visión Global del Trabajo Práctico	1
Sobre la Simulación.....	1
Sobre los TADs	2
El TAD Actividad.....	2
El TAD CalendarioAcademico	2
El TAD Curso	3
El TAD MesaDeExamenFinal	3
Cronograma.....	3
Forma de Entrega	3
Herramientas y Métodos para la Construcción.....	3

Objetivos

Adquirir conocimientos y práctica sobre

- Especificación de TADs
- Implementación de TADs en ANSI C
- Desarrollos en ANSI C: Lenguaje y Biblioteca Estándar
- Aplicación de AFs

Medios para Cumplir estos Objetivos

Un trabajo práctico donde se aplican los conceptos.

Este trabajo práctico será dividido en varias etapas, más una de integración. Cada etapa contendrá la especificación e implementación de TADs y las pruebas sobre sus implementaciones. Estas pruebas deben verificar la consistencia entre cada implementación y su especificación previa.

Contexto del Trabajo Práctico -- Simulación de un Año Académico de SSL

El TP consistirá de varias entregas, las cuales girarán en torno a la Simulación de una Año Académico de SSL. Para ello, se deberán diseñar los TAD: 1. **Actividad**; 2. **CalendarioAcademico**, que contiene valores del TAD Actividad; 3. **Curso** y 4. **MesaDeExamenFinal**. Además, se realizará un programa que realice la **simulación** valiéndose de estos TADs.

Visión Global del Trabajo Práctico

Sobre la Simulación

El diseño de la simulación se basará sobre un **modelo simplificado** de un año académico de SSL.

La simulación será de las actividades de un curso del primer cuatrimestre y uno del segundo cuatrimestre (con sus respectivos inscripciones y cierres), como así también de los exámenes finales durante el año.

La simulación tendrá como eje central y elemento definitorio al **calendario académico**. En el calendario académico se establece, día a día, las **actividades** programadas para SSL; por ejemplo: clase normal, examen final y cierre de curso.

El programa simulador tendrá como entrada principal el calendario, pero también necesitará de **los listados de alumnos** que se desean inscribirse a los **cursos** y a los **exámenes finales**.

En cuanto a las salidas, se generarán los **nuevos listados de alumnos en condición de inscribirse a cursos o a examen final** en función de cómo rindieron en la cursada ó examen. También se generarán las **actas de regularización de cursada y de examen final**.

Toda simulación tiene elementos **probabilísticos**. En este caso, se deben manejar la probabilidad de que cada alumno apruebe una instancia de evaluación, desee inscribirse a un examen final, etc.

Conceptos de ANSI C necesarios

- Entrada / Salida Estándar (stdio.h).
- Flujos de texto y binarios.
- Funciones.

Sobre los TADs

La especificación se podrá inferir de los enunciados de las entregas, pero ciertas decisiones de diseño del TAD serán tomadas por cada equipo. Cada grupo deberá completar los puntos indispensables para completar la especificación, que no estén incluidos en el enunciado.

Los TADs se implementarán en **ANSI C**. Cada TAD será encapsulado dentro de una **Biblioteca**. Para generar la biblioteca se deberá diseñar y construir la parte **pública** ó **interfase** de la biblioteca en un archivo header (encabezado) con extensión .h y la parte **privada** o **implementación** en un archivo con extensión .c.

Cualquier valor de un TAD puede representarse como un literal. No es posible extender el lenguaje de **literales** (constantes) de ANSI C, por lo que para cada TAD se implementará una función que, dada una cadena que representa un literal, retorna un valor del TAD. Por ejemplo: el lenguaje ANSI C puede representar números enteros en base 8, 10 y 16 (ver [M3]), pero no tiene la capacidad de representar a los enteros cómo números romanos. No es posible la siguiente inicialización:

```
int i = XVIII;
```

pero, mediante la construcción de una función, sí es posible la siguiente inicialización:

```
int i = Entero_LiteralRomano("XVIII");
```

El TAD Actividad

El TAD **Actividad** representa una **actividad**, **evento** ó **suceso** relacionado con SSL que se desarrolla en una fecha específica del calendario académico. Las actividades tienen una **fecha** del calendario académico, un **tipo** y una **descripción**.

Para simplificar la implementación de este TAD, informamos la forma de representar los componentes de los valores del TAD. El conjunto de valores que representan los tipos de actividades serán implementados con un *enum*. Se utilizará un *struct* para la representación de la actividad. El componente fecha será representado con un número natural entre 1 y 366.

Las operaciones de este TAD incluyen: las de comparación; las de obtención de día, mes, descripción y tipo; y predicados como EsDiaConActividad.

Conceptos de ANSI C necesarios

- Tipos de datos de primitivos.
- Cadenas.
- Estructuras (struct).
- Enumeraciones (enum).
- Funciones.

El TAD CalendarioAcademico

El TAD **CalendarioAcademico** define el **cronograma** de **actividades** relacionadas con SSL que se realizarán a lo largo de un **año académico**.

Cada año académico comienza el primero de Abril de un determinado año y finaliza el 31 de Marzo del año siguiente, el calendario se divide en dos cuatrimestres, de 16 semanas cada uno.

A grandes rasgos, un valor del TAD CalendarioAcademico es un **año académico**, una **secuencia finita de valores del TAD Actividad** y un indicador de la **actividad actual** que servirá para implementar la simulación.

Las operaciones de este TAD son las que permiten recorrer todo el calendario, como, por ejemplo, las operaciones a las que les damos los siguientes nombres: SetDiaActual, GetSiguienteDiaConActividades, GetDiaActual, GetActividadActual; y la que permite añadir una actividad al calendario: AddActividad.

Conceptos de ANSI C necesarios

- Arreglos.
- *Alocación* dinámica de memoria (malloc).
- Y los conceptos enunciados anteriormente.

El TAD Curso

EL TAD **Curso** representa un conjunto de **alumnos** que cursa SSL durante un **cuatrimestre** determinado y con un **Docente** determinado.

Cursar implica **asistir** y **rendir evaluaciones**, por lo que un curso debe contener esa información. Las 32 clases del cuatrimestre se representarán en 4 bytes (32 bits) para cada alumno.

Para que un alumno pueda **regularizar** su cursada, deberá asistir a un **mínimo de clases** y **aprobar** todas las instancias de evaluación.

Las operaciones del TAD curso incluyen: AbrirCurso, AddAlumno, CerrarCurso, TomarAsistencia, TomarEvaluacion, EmitirTpa, EmitirAlumnosRecursantes, EmitirAlumnosEnCondicionDeRendirFinal.

Conceptos de ANSI C necesarios

- Operadores de bits (>> y <<), para la asistencia de cada alumno.
- Generación de números pseudo aleatorios.
- Entrada / Salida Estándar (stdio.h).
- Flujos de texto y binarios.
- Y los conceptos enunciados anteriormente.

El TAD MesaDeExamenFinal

El TAD **MesaDeExamenFinal** representa un conjunto de **alumnos** que rinden **examen final** de SSL en un determinado día, con una mesa examinadora de tres **profesores**.

Las operaciones incluyen: AbrirMesa, AddAlumno, CerrarMesa, TomarEvaluacion, EmitirActa, EmitirAlumnosAprobados, EmitirAlumnosDesaprobados, EmitirAlumnosAusentes.

Conceptos de ANSI C necesarios

- Los conceptos enunciados anteriormente.

Cronograma

Entregas de las Resoluciones por parte de los Alumnos

- Semana del **16 de Mayo** Entrega #1 – TAD **Actividad**
- Semana del **23 de Mayo** Entrega #2 – TAD **MesaDeExamenFinal**
- Semana del **06 de Junio** Entrega #3 – TAD **Curso**
- Semana del **20 de Junio** Entrega #4 – **Simulación** y TAD **Calendario**

Evaluaciones

- Semana del 27 de Junio **Coloquio** con defensa del TP.
- Semana del 4 de Julio (última semana de clase) **1er recuperatorio** de Coloquio con defensa del TP.
- Semana del 11 de Julio **2do recuperatorio** de Coloquio con defensa del TP.

Forma de Entrega

Referirse al TP #0.

Herramientas y Métodos para la Construcción

Referirse al TP #0.