

Anuncios

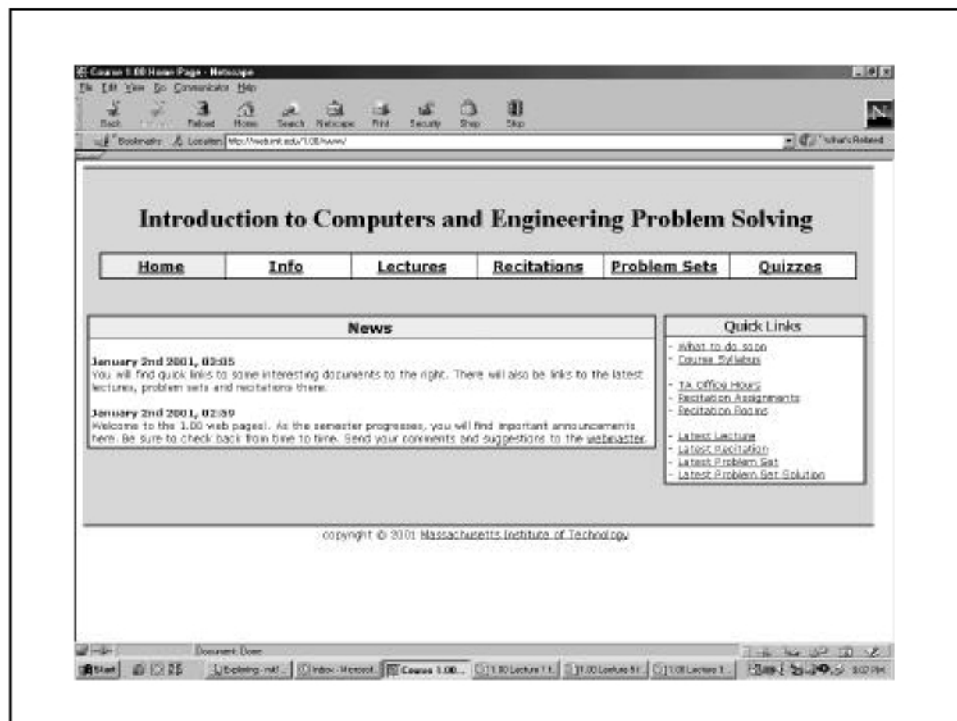
- **Materiales de entrega (5):**
 - Programa del curso.
 - Solicitud de inscripción para el seminario / clase práctica (hasta las 16:00 de hoy).
 - Apuntes de la clase 1.
 - Tarea 1.
 - Valoración inicial:
 - Por favor, háganlo ahora y entréguenlo al salir de clase.
 - No se le asignará nota; sirve únicamente como complemento para la evaluación del curso.

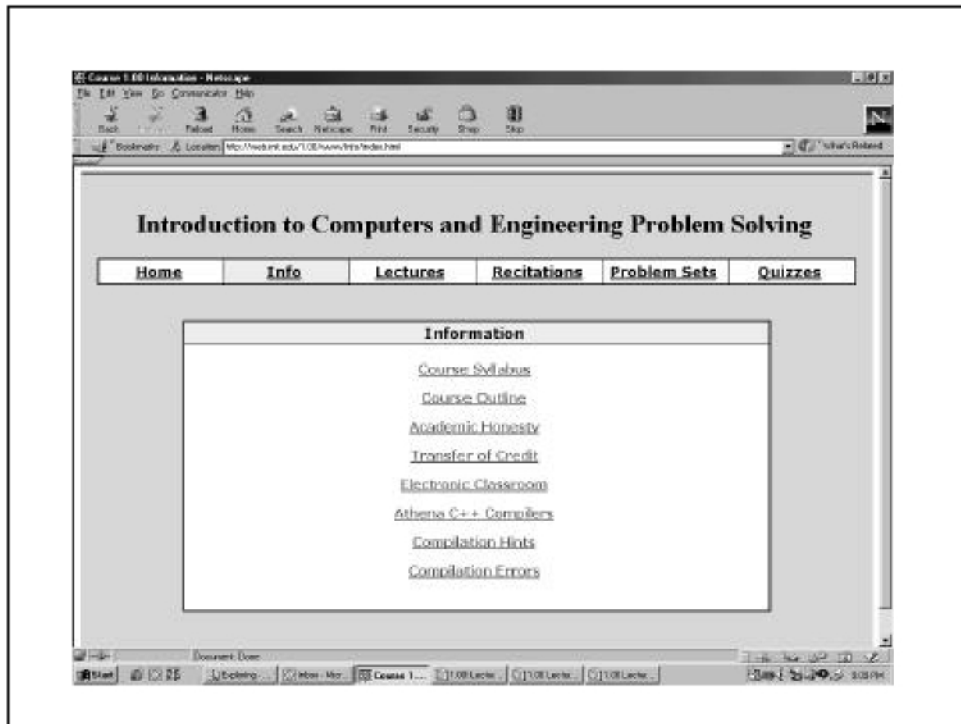
1.00 Clase 1

**Visión global del curso
Introducción a Java**

Información sobre el curso 1.00

- Personal docente:
 - 2 profesores, 8 ayudantes técnicos, 2 ayudantes de investigación (RAs), 2 monitores de prácticas y los examinadores.
- Página Web del curso:
 - web.mit.edu/1.00/www/
 - Toda la información sobre el curso está disponible en la Web.
 - Las clases, las prácticas, los seminarios y los boletines de problemas se publicarán el mismo día que tengan lugar.
 - Las fotocopias para la clase, práctica o seminario:
 - Recójalas al entrar.
- Estudiantes de postgrado: matricúlense en el curso 1.001, no en el 1.00.





Objetivos del curso

- **Conceptos clave sobre el desarrollo de software:**
 - Diseño de software (y un poco sobre los requisitos).
 - Desarrollo y depuración / pruebas.
 - El trabajo en equipo en la implementación de software
- **Programación en un entorno interactivo y orientado a objetos:**
 - Java; el C# de Microsoft es muy parecido.
 - Introducción muy breve a C++ al final; también similar.
- **Uso de la computación para solucionar problemas científicos, de ingeniería y de gestión:**
 - Los ejercicios cubren una gran variedad de problemas.
- **Patrones de software, p. ej.:**
 - Modelo-vista-controlador.
 - Generador del problema- solucionador- generador de la salida.

Objetivos del curso, 2ª parte

- **Interfaces gráficas de usuario:**
 - Paquete Swing de Java y modelos de eventos.
- **Algoritmos:**
 - Ordenación, búsqueda, recorrido (*traversing*)...
 - Conceptos, programación y “librerías”.
- **Estructuras de datos:**
 - Pilas, colas, árboles, listas...
 - Conceptos, programación y librerías.
- **Uso de bibliotecas:**
 - Módulos preestablecidos para tareas comunes.

Portátiles, prácticas y seminarios

- **Entregue al final de la clase su solicitud de inscripción para el seminario / clase práctica y para la obtención del portátil. También puede entregarla hoy a las 16:00 h. en el aula 5-336.**
 - Su compañero de portátil y los trabajos de la clase práctica y el seminario, aparecerán anunciados en la Web mañana a las 9:00 h.
- **Recoja el portátil mañana (jueves) o el viernes.**
 - Aula 26-055: jueves, de 9:00 a 17:00 h.; viernes, de 9:00 a 13:00 h.
 - Cada pareja de estudiantes recibirá un portátil para el trimestre.
 - Su compañero debe estar presente en la recogida del mismo.
- **La primera clase práctica es el viernes: de 1 a 2:30 ó de 3 a 4:30 en el aula 26-152.**
 - La clase práctica tendrá lugar cada viernes durante todo el trimestre en ese horario (2 grupos).
 - Asistencia obligatoria. Venga con su compañero y traigan el portátil.
- **Los seminarios comienzan el lunes y martes de la próxima semana.**
 - Grupos separados: con y sin experiencia.
 - Asistencia obligatoria. Venga con su compañero y traigan el portátil.

Portátiles, prácticas y seminarios (2ª parte)

- En las prácticas del viernes trabajaremos con el portátil y realizaremos ejercicios de aprendizaje activo:
 - Clases cortas con ejercicios sobre las prácticas: programación, simulaciones y ejercicios breves.
 - Los ayudantes técnicos y los profesores prestarán ayuda durante las prácticas.
- En los seminarios también se verán métodos de aprendizaje activo:
 - 8 estudiantes por seminario (4 parejas).
 - Preguntas breves, revisión de temas de clase y ejercicios de diseño.
- Portátiles:
 - Los portátiles poseen tarjetas de red inalámbricas. Se pueden utilizar por todo el campus.
 - Puede utilizarlo también para otras clases durante este trimestre.
 - Devuélvalo el día del examen final o antes.
- Tareas:
 - Entréguelas a través de *Command* (command.mit.edu).
 - Regístrese en *Command* durante los próximos días.

Cómo escribir programas en Java

- Ordenadores portátiles (*Microsoft Windows*)
 - Entorno de desarrollo integrado (IDE) Forte para Java.
 - Puede cargar Java y Forte en el portátil o en su equipo de sobremesa:
 - Muy recomendado Windows 2000 y 256MB de RAM.
 - Descárgue las instrucciones para Java y el IDE Forte que están publicadas en la página Web del curso 1.00.
 - En la clase práctica del viernes y en los seminarios de la próxima semana, se le enseñará a utilizar el IDE Forte.
- Terminales de trabajo Athena (UNIX)
 - También disponible el IDE Forte para Java; el mismo que en los portátiles.

Requisitos del curso y recursos

- **Requisitos del curso:**
 - 10 boletines de problemas (60% de la nota). Se entregarán los viernes.
 - 2 pruebas (16%). Se realizarán los miércoles en horario normal de clase. No habrá pruebas por la tarde este trimestre.
 - Un examen final durante el periodo de los finales (17%).
 - En las pruebas y en el examen se permite el uso de libros y apuntes.
 - Seminarios semanales. Asistencia obligatoria (3%).
 - Prácticas semanales cada viernes. Asistencia obligatoria (4%).
- **Recursos para el curso:**
 - Horarios de consulta del ayudante técnico: +16 horas a la semana. El horario se publicará en la Web y en el aula 5- 336 el próximo lunes.
 - Horarios de los monitores de prácticas: miércoles y jueves por la tarde, 8 horas cada día.
 - Horario de consultas del profesor después de clase y por cita previa.
 - Libro: Horstmann/Cornell, *Core Java*, Vol. 1

Honestidad académica

- Puede ayudar a un compañero en la comprensión de las clases, de las prácticas, del libro, de los seminarios y de los enunciados de los problemas.
- Puede debatir sobre el diseño de su programa: opciones para las clases, firmas de los métodos.
- **Pero luego debe escribir por sí mismo su código en Java.**
- Al escribir sus programas, podrá obtener ayuda de otros estudiantes sólo para:
 - Que le ayuden a encontrar un error, pero no a corregirlo.
 - Que le expliquen la sintaxis de Java. Utilice un ejemplo que no sea el programa que está escribiendo.
- Es preferible que, cuando esté escribiendo el programa, pida ayuda a los ayudantes técnicos y a los profesores.

Iniciativa del portátil con tarjeta de red inalámbrica

- El curso 1.00 es uno de los 4 proyectos pilotos en los que se entregan portátiles con tarjeta de red inalámbrica.
- ¿Por qué portátiles? Algunas razones:
 - Acceso práctico y sencillo a la informática.
 - Valorar el aprendizaje colaborador.
 - Analizar la compatibilidad de esta tecnología.
 - Determinar los atributos tácitos de aprendizaje.

¿Qué obtendrá?

- Un equipo compacto HP Omnibook 6000 con:
 - 256MB de RAM.
 - 30 GB de disco duro.
 - Tarjetas LAN inalámbricas *Lucent* (802.11b).
 - Pentium III a 650 MHz .
 - Una batería adicional.
 - Un maletín.
 - Software: Forte, Netscape, TSM (copias de seguridad) y otras utilidades del MIT.

Cómo obtener ayuda

- **Apoyo técnico del curso: Bassam Chaptini, para todas las cuestiones relacionadas con el curso.**
 - 3-3456, bhc@mit.edu, 9-355 (Carrel J)
Para servicio técnico de hardware en general y otras aplicaciones del MIT, póngase en contacto con Bassam o con los I/S.
 - Servicio técnico de portátiles: # 452-4757 (2-4SLP), o laptop-project@mit.edu

Sus responsabilidades

- **Practique la “informática segura”.**
 - El uso promiscuo exige cuidado.
- **Proporcione un buen cuidado a su portátil.**
- **Al final del semestre, devuelva el ordenador en buenas condiciones.**
- **Si ocurriera lo impensable ...**
 - Póngase en contacto con la policía del campus para denunciar el robo.
 - Notifíquelo inmediatamente a su profesor o a algún contacto técnico del curso.

Nuestras responsabilidades hacia usted

- Proporcionar servicio técnico para las aplicaciones y la conectividad del MIT.
- Reparación del hardware: servicio de préstamo temporal de ordenadores en caso de que el del usuario presente algún problema.
- Asistencia en general.

Responsabilidades mutuas

- Realizar copias de seguridad:
 - Se ha instalado TSM, *Tivoli Storage Manager* (gestor de almacenamiento).
 - *Es necesario que lo utilice.*
- Restauración:
 - En el peor de los casos: el I/S realiza una restauración del sistema y devuelve la máquina a su estado original.
 - Cargue sus datos a partir de las copias de seguridad del TSM o desde su directorio personal AFS (sistema de archivos en red al que se puede acceder desde cualquier máquina del país que tenga instalado el AFS o Sistema de archivos de Andrew).
- ¡Convierta esto en un proyecto piloto con éxito!
 - Préstenos su ayuda, ofreciéndose voluntario para llevar un diario de sus experiencias, y participe en grupos de estudio (3) durante el transcurso del semestre (¡pizza gratis!!!).

Java

- **Lenguaje de programación moderno utilizado en aplicaciones técnicas y comerciales:**
 - Orientado a objetos, al igual que la mayoría del software moderno.
 - Centrado en Internet, respalda la computación distribuida.
- **En la primera mitad del curso, estudiaremos los elementos básicos de Java:**
 - Nos centraremos en la sintaxis, en los conceptos básicos y en la interfaz de usuario.
- **En la segunda mitad del curso, estudiaremos aplicaciones orientadas a objetos:**
 - Nos centraremos en aplicaciones numéricas, estructuras de datos y algoritmos.

Programación orientada a objetos

- **Los objetos son cosas (“entidades”) que poseen estados (campos de datos) y comportamientos (métodos, funciones):**
 - Constituyen un modo de organizar extensos programas en partes comprensibles, mantenibles y reutilizables.
 - Los programas del curso 1.00, a excepción de la tarea 1, consistirán en un conjunto de objetos que interactúan entre sí para producir los resultados deseados.
 - Los ejemplos serán tuberías con flujos de fluidos, itinerarios de autobuses en redes de autobuses, polinomios, robots y dispositivos elásticos de embalaje, diccionarios de palabras mal escritas, etc.
- **Clases y patrones a partir de los cuales se construyen los objetos: clase Tubería -> muchas Tuberías (objetos).**

Programación orientada a objetos

- La comunicación entre objetos se realiza a través del paso de mensajes:
 - Invocan comportamientos (métodos) y pasan parámetros (datos) en los mensajes.
- Los objetos encapsulan u ocultan información:
 - Los detalles de un objeto se ocultan para no ser vistos por otros objetos. De esta forma, los detalles de los objetos en general, no tienen por qué ser conocidos entre sí.
- “El método *Main*” lanza los objetos y poco más (excepto en el ejercicio 1).
- Los objetos se pueden extender o ampliar a través de mecanismos de herencia:
 - Los niños poseen rasgos de sus padres (estado y comportamiento) y pueden modificarlos o añadir más.
 - Los objetos pueden invocar dinámicamente a objetos que no existían (aún no se habían escrito) cuando el invocador se escribió. Este y otros conceptos acerca de los objetos, fomentan la reutilización del código.

Cómo desarrollar un programa en Java

- **Lea la tarea y compréndala:**
 - Si no sabe lo que tiene que hacer, no será capaz de hacerlo.
- **Esquematice un diseño: objetos, estado, comportamiento:**
 - Decida cómo va a enfocar el problema.
 - Esquematice el enfoque con palabras o dibujos. Organícelo en fases.
- **Escriba el programa en Java, utilizando Forte:**
 - Cree archivos de código fuente de Java en el explorador de proyectos de Forte.
 - Escriba el código de Java utilizando el editor de Forte.
 - Escriba sólo lo que crea que se compilará en cada fase (p. ej., la lectura de la entrada). Utilice el compilador de Forte para Java.
 - Una vez compilada una fase, escriba y compile la siguiente. El tamaño de las fases aumentará a lo largo del trimestre.
- **Realice pruebas, leyendo y revisando principalmente el código en Forte:**
 - Utilice el depurador de Forte para leer el código.
- **Repita el ciclo de nuevo para recopilar detalles.**

Ejemplo de aplicación en Java

```
public class Welcome1 {
    public static void main(String[] args) {
        System.out.println("Bienvenido al 1.00"); int
        estudiantes= 240; int licenciados= 35;
        double pctLic= (double) licenciados/estudiantes;
        Systemt out.println("Porcentaje de Licenciados: " + pctLic);
        System.exit(0);
    }
}

// En las transparencias de clase se utilizará formato comprimido con {}
// Deje más espacio en blanco en su código
// En las transparencias de clase se omitirá System.exit(0); debe utilizarlo
```

Ejemplo de aplicación de interfaz gráfica de usuario (GUI)

```
// La aplicación GUI abre su propia ventana (marco) en el PC
import javax.swing.*;
import java.awt.*;

public class Welcome extends JFrame {
    // Crea una nueva forma (objeto) Welcome
    // main es invocado cuando la aplicación comienza
    public static void main(String args[]) {
        Welcome app= new Welcome();
        app.setDefaultCloseOperation(EXIT_ON_CLOSE);
        app.show();
    }

    public Welcome() {          // Constructor invocado en la creación

        JLabel myLabel= new JLabel("Bienvenido una vez más al 1.00");
        setSize(300,200);
        Container conPane= getContentPane();
        conPane.add(myLabel);
    }
}
```

Lo que hay que hacer

- **Entregue su valoración inicial.**
- **Entregue hoy su solicitud para la asignación al seminario sobre las 16:00 h. en el aula 5-336.**
 - Los que necesiten un compañero: por favor, reúnanse un momento a la entrada o fuera y encuentren uno ahora.
- **Mañana a las 9:00 h. en la página Web del curso 1.00, compruebe sus grupos de seminario y clases prácticas, y confirme que tiene compañero.**
- **Recoja su portátil mañana o el viernes.**
 - Ambos compañeros deben estar presentes en la recogida.
- **Asista el viernes al grupo de prácticas que se le ha asignado.**
 - Venga con su compañero y traigan el portátil; les enseñaremos a utilizar Forte.
- **Los seminarios comenzarán el lunes y martes de la semana próxima.**
- **Regístrese.**