Ministerio de Cultura y Educación
Universidad Nacional de San Luis
Facultad de Ciencias Físico Matemáticas y Naturales
Departamento: Informatica
Area: Area IV: Pr. y Met. de Des. del Soft.

(Programa del año 2015)
(Programa en trámite de aprobación)
(Presentado el 10/04/2015 11:27:09)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
METODOS FORMALES	ING. INFORM.	026/12-08/15	2015	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
RIESCO, DANIEL EDGARDO	Prof. Responsable	P.Adj Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	**Teóricas**	**Prácticas de Aula**	**Práct. de lab/ camp/ Resid/ PIP, etc.**	**Total**
Hs	2 Hs	1 Hs	2 Hs	5 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1º Cuatrimestre

Duración			
Desde	**Hasta**	**Cantidad de Semanas**	**Cantidad de Horas**
16/03/2015	26/06/2015	15	75

IV - Fundamentación

Las computadoras están siendo empleadas para múltiples tareas donde la presencia de fallas puede traer aparejado severas consecuencias no sólo a nivel de pérdidas económicas sino de vidas humanas.
Esta tendencia se está incrementando cada vez más. Las computadoras juegan un rol esencial en el control aeroespacial, en la aeronavegación, en el control de trenes, autos, reactores nucleares y equipamiento hospitalario, por dar simplemente algunas pocas de las aplicaciones más críticas. Si las personas están preocupadas por un aterrizaje seguro de su avión o que su cuenta bancaria tenga el saldo correcto, entonces concordarán que es de vital importancia que los sistemas de computación que están por detrás de ellos sean altamente confiables.
Para construir software de alta calidad, sin descartar otras alternativas, una de las vías más importantes es definiendo el dominio y los requerimientos a través de especificaciones formales.

V - Objetivos / Resultados de Aprendizaje

El objetivo de la materia es que los alumnos tengan la posibilidad de incursionar en los métodos formales como herramienta de producción de software de alta calidad.

VI - Contenidos

Unidad I: Introducción
Introducción al curso y a los Métodos Formales. El papel de los métodos formales en ingeniería de software. Pro y contra de

los Métodos Formales. Un panorama de los métodos formales. Diversas clasificaciones de los Métodos Formales. Estilos de especificaciones. Estilos de desarrollo. Grados de rigor. Algunas aplicaciones industriales.

Unidad II: Especificaciones formales
Enfoques de especificación de un sistema. Especificaciones basadas en Lógica de Primer Orden y Teoría de Conjuntos. Especificaciones algebraicas. Especificaciones orientadas a la propiedad. Especificaciones orientadas al modelo. Especificaciones aplicativas, imperativas, concurrentes. Lightweight Formal Methods. Estilos de especificaciones en OBJ, Larch, RAISE, Z, VDM, B, PVS.

Unidad III: Verificación formal de sistemas de software y de sus propiedades
Verificación formal del software. Sistema formal. Sintaxis, semántica y sistema de prueba. Verificación de propiedades de corrección, terminación, refinamiento. Relación de refinamiento. Stepwise refinement. Refinamiento de funciones. Refinamiento de datos.

Unidad IV: El Método RAISE
Introducción a RAISE. Características generales de RSL. Expresión básica de clase. Definiciones de tipos. Expresiones de tipo: built-in y definidos por el usuario. Sorts. Constructores de tipo. Definiciones de values: funciones y constantes. Estilos: explícito e implícito. Definiciones de axiomas. Chaos. Lógica condicional de RAISE. Módulos.

Unidad V: Otros estilos de especificaciones formales.
Especificación de comportamiento. Métodos basados en álgebra de procesos. Interacción entre procesos concurrentes. Métodos basados en Redes de Petri. Formalización de eventos concurrentes. Métodos basados en lógica temporal.

VII - Plan de Trabajos Prácticos

1. En aula: Expresión básica de clase. Definiciones de tipos. Expresiones de tipo: built-in y definidos por el usuario. Sorts. Constructores de tipo. Definiciones de values: funciones y constantes. Estilos: explícito e implícito.
En laboratorio: de los temas del práctico de aula.

2. En aula: Definiciones de axiomas. Chaos. Lógica condicional de RAISE. Expresiones if.
En laboratorio: de los temas del práctico de aula.

3. En aula: Producto Cartesiano. Funciones. Funciones totales y parciales. Conjuntos finitos e infinitos. Listas finitas e infinitas. Mapas finitos determinísticos y mapas parciales.
En laboratorio: de los temas del práctico de aula.

4. En aula: Expresiones de subtipos. Subtipos vs. axiomas. Tipos maximales. Subtipos vacíos.
En laboratorio: de los temas del práctico de aula.

5. En aula: Variants. Constructores de constantes. Constructores de records. Destructores y reconstructores. Uniones disjuntas de tipos. Definiciones de short records.
En laboratorio: de los temas del práctico de aula.

6. En aula: Expresiones Case y Let. Forma general de una expresión CASE. Uso. Patrones: Literales, Wildcard, de Productos, de Nombres, de Records, de Listas. Forma general de expresiones Let implícitas y explícitas. Uso. Equivalencias entre Let y Case. Anidamiento.
En laboratorio: de los temas del práctico de aula.

7. En aula: Especificaciones concurrentes en RSL.
En laboratorio: de los temas del práctico de aula.

8. En laboratorio: Especificación de un caso real.

VIII - Regimen de Aprobación

La materia se desarrolla con la modalidad de promoción sin examen final. Existen dos niveles:

a)Regularización solamente: Para regularizar la materia se deberá:
1. Tener como mínimo un 80% de asistencia a clases prácticas.
2. Tener los prácticos, solicitados por la cátedra, aprobados.
3. Aprobar un parcial o su recuperación con una nota mínima de seis (6) sobre diez (10).

b) Promoción sin examen final: Para regularizar y aprobar la materia se deberá:
1. Cumplir con los requisitos a.1 y a.2.
2. Aprobar el parcial o su recuperación con una nota mínima de siete (7) sobre diez (10).
3. Aprobar una Evaluación Global Integradora con con una nota mínima de siete (7) sobre diez (10).

Aquellos alumnos que sólo regularicen la materia deberán rendir un examen final, en los turnos establecidos.
No se admitirán exámenes de alumnos libres.

Se otorga, tal como lo expresa la reglamentación vigente, una recuperación adicional.

IX - Bibliografía Básica

[1] - Alagar, V. & Periyasamy, K. "Specification of Software Systems". Springer-Verlag New York, Inc. 1998.
[2] - Bibliografia sobre PVS puede encontrarse en el sitio oficial de PVS: http://pvs.csl.sri.com/
[3] - Bowen, J., & Hinchey, M., "Seven more myths of formal methods". IEEE Software, 12(3). Julio, 1995.
[4] - Bowen, J., & Hinchey, M., "Ten commandments of formal methods". IEEE Computer. Abril, 1995.
[5] - C. A. R. Hoare. "Communicating Sequential Processes", Prentice Hall, ISBN 0-13-153289-8. Este libro básico fundamental del tema ha sido actualizado por Jim Davies (Oxford University Computing Laboratory) y la nueva edición puede obtenerse en formato PDF en http://www.usingcsp.com/cspbook.pdf.
[6] - Dasso, A., Funes, A., "Formal Methods in Software Engineering" en Encyclopedia of Information Science and Technology, Idea Group, 2005.
[7] - Funes, A., George, C., "Formalizing UML class diagrams", capítulo 8 de UML and the Unified Process, Idea Group, 2003.
[8] - Hung Dang Van, Chris George, Tomasz Janowski, and Richard Moore (Eds). "Specification Case Studies in RAISE". FACIT sries. Springer-Verlag. 2002.
[9] - J. L. Peterson, "Petri Net Theory and the Modeling of Systems", Prentice-Hall, N.J., 1981, ISBN: 0-13-661983-5
[10] - J.C.M. Baeten. "A Brief History of Process Algebra". Report CS-R 04-02, Department of Mathematics and Computer Science, Technische Universiteit Eindhoven, http://www.win.tue.nl/fm/pubbaeten.html.
[11] - Jean-Raymond Abrial. "The B-Book: Assigning Programs to Meanings". Cambridge University Press, 1996. ISBN 0-521-49619-5.
[12] - Jonathan Bowen. "Formal Specification and Documentation using Z: A Case Study Approach". International Thomson Computer Press (ITCP) Thomson Publishing. ISBN 1850322309.
[13] - Luqi & Goguen, J., "Formal methods: Promises and problems", IEEE Software, 14(1), págs. 73-85., Ene-Feb 1997.
[14] - Michael Huth and Mark Ryan. "Logic in Computer Science: Modelling and Reasoning about Systems". Cambridge University Press, 2000.
[15] - Steve Schneider. "The B-Method: An Introduction". Palgrave, Cornerstones of Computing series, 2001. ISBN 0-333-79284-X.
[16] - The RAISE Language Group. "The RAISE Specification Language". Prentice-Hall International, 1992.
[17] - The RAISE Method Group. "The RAISE Development Method". Prentice-Hall International, 1995.
[18] - W. Reisig, "Petri Nets, An Introduction", EATCS, Monographs on Theoretical Computer Science, W.Brauer, G. Rozenberg, A. Salomaa (Eds.), Springer Verlag, Berlin, 1985.
[19] - Bibliografia adicional puede obtenerse del sitio http://formalmethods.wikia.com/wiki/Formal_Methods_Wiki.

X - Bibliografia Complementaria

[1]

XI - Resumen de Objetivos

Incursionar en los métodos formales como herramienta de producción de software de alta calidad.

XII - Resumen del Programa

Unidad I: Introducción
Unidad II: Especificaciones formales
Unidad III: Verificación formal de sistemas de software y de sus Unidad IV: El Método RAISE
Unidad V: Otros estilos de especificaciones formales.

XIII - Imprevistos

.

XIV - Otros

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
	Profesor Responsable
Firma:	
Aclaración:	
Fecha:	