

UNIVERSIDAD DE BUENOS AIRES

FACULTAD DE CIENCIAS EXACTAS Y NATURALES



DEPARTAMENTO: Computación.....

ASIGNATURA: .. [REDACTED] .....

CARRERA/S:..Licenciatura en Cs. de la Computación.....

CARACTER:..obligatoria....(indicar si es obligatoria u optativa)

PUNTAJE:....-----.....(en caso de ser optativa)

DURACION DE LA MATERIA:..cuatrimestral.....(indicar si es cuatrimestral o anual).

HORAS DE CLASE: a) TEORICAS....4. HS. b) PROBLEMAS ..... HS.  
c) LABORATORIO.6. HS. d) SEMINARIOS..... HS.  
e) TOTALES.....10 HS.

ASIGNATURAS CORRELATIVAS:..Organización del  
Computador;..Logica;.....  
..Laboratorio II.....

PROGRAMA:

1.-Un lenguaje para especificar acciones y una metodología para desarrollar simultáneamente los problemas con su prueba de corrección: Precondición más débil. La precondición más débil como semántica de un pequeño lenguaje de programación. Comando skip. La precondición más débil de la composición. Asignación. Alternativa. Teorema de la alternativa. Construcción de programas aplicando el Teorema de la alternativa y asignaciones. Iteración. Construcción de invariantes y funciones de cota a partir del estado final. Construcción de ciclos a partir del invariante y la cota. Algoritmos de ordenamiento como caso de estudio (inserción, selección).

2.-Precondición más débil de la llamada a procedimiento. Teoremas útiles. La recursión. Dividir para conquistar: Algoritmo de ordenamiento como caso de estudio (megersort, quicksort). Backtracking.

3.-Tipos abstractos pila, cola, árbol, cola de prioridad. Especificación de problemas mediante tipos abstractos. El Problema del diccionario.

4.-Implementación de tipos abstractos. La función de abstracción. Prueba de corrección de una implementación. Implementación de las funciones mediante acciones, usando precondición más débil. Diversas implementaciones de diccionario. Nociones elementales de adecuación de una implementación: algoritmos de ordenamiento (selección vs. heapsort) como caso de estudio; backtracking como caso de estudio; representación del estado.



Bibliografía:

A. Aho, J. Hopcroft y J. Ullman, "Data Structures and Algorithms"  
Addison-Wesley, 1983.

R. Bird y Ph. Wadler, "Introduction to Functional Programming".  
Prentice-Hall, 1988.

E. Dijkstra, "A Discipline of Programming".  
Prentice-Hall, 1976.

E. Dijkstra y W. Feijen, "A Method of Programming".  
Addison-Wesley, 1988.

D. Gries, "The Science of Programming".  
Springer-Verlag, 1985.

J. Guttag y J. Horning, "The Algebraic Specification of Abstract  
Data Types".  
Acta Informatica 10, pp. 27-52, 1978

D. Knuth, "The Art of Computer Programming", Vol. 1 y 3.  
Addison-Wesley, 1975.

H. Partsch, "Specification and Transformation of Programs".  
Springer-Verlag, 1990.

R. Sedgewick, "Algorithms".  
Addison-Wesley, 1989.

N. Wirth, "Algoririthmis + Data Structures = Programs".  
Prentice-Hall, 1976.

*Rosa Wachencrauzer*  
ROSA WACHENCRAUZER

*Y. LOISEAU*  
DIRECTORA  
Depto. de Computación  
F.C.E. y N. - U.B.A.