Abstracts

Approximating pre-emptive priority dispatching in a multiprogramming model, H. A. Anderson, Jr., *IBM Journal of Research and Development* 17, No. 6, 533-539 (November 1973). The formulation of the closed queuing network model of a multiprogramming computer system is generalized to allow each task to have its own set of facility service rates and I/O device selection probability distribution. In the model, processor sharing is assumed for different types of customers. It is shown through a series of investigations that the model reasonably approximates pre-emptive priority dispatching.

Economic order and surplus quantities model, M. N. El Agizy, *IBM Journal of Research and Development* 18, No. 1, 72-75 (January 1974). A standard mathematical model for inventory management is known as the Economic Order Quantity (EOQ) model. In this communication the EOQ model is extended to include the possibility of determining how much, if any, excess stock should be sold at the beginning of a decision period. The new model is of practical importance for situations in which a formal inventory management system is to be instituted while substantial inventories exist or when changes in demand, ordering cost, or carrying and interest charges require recomputation of the economic order quantity.

Floating-point computation, P. H. Sterbenz, Prentice-Hall, Inc., Englewood Cliffs, New Jersey (1974). This book presents floating-point computation from the point of view of a computer user who is writing programs in a higher-level language such as FORTRAN or PL/I and is concerned about the quality of the answers he produces. It discusses what actually happens when floating-point arithmetic is performed during the execution of his program and how the nuances of floating-point arithmetic affect the answer. The arithmetic of the IBM System/370 is stressed, but many of the results are generalized to apply to other machines.

Implementation of a high level language machine, A. Hassitt, J. W. Lageschulte, and L. E. Lyon, *Communications of the ACM* 16, No. 4, 199-212 (April 1973). Computing machines which directly execute the statements of a high-level language have been proposed in the past. This report describes the actual implementation of such a machine: it is a computer whose "machine language" is APL. The machine is fully operational and correctly executes almost all of the APL operations on scalars, vectors, and arrays. The machine automatically allocates memory, executes statements, calls functions, converts numbers from one type to another, checks subscripts, and automatically detects many types of programmer errors.

Improved modeling of computer hardware systems, H. D. Caplener and J. A. Janku, *Computer Design* 12, No. 8, 59-66 (August 1973). Step-by-step computer simulation of digital hardware as it is designed allows its internal performance as well as its interaction with other systems to be predicted before any hardware is built.

Placement of records on a secondary storage device to minimize access time, D. D. Grossman and H. F. Silverman, *Journal of the Association for Computing Machinery* 20, No. 3, 238-429 (July 1973). The problem considered is how to place records on a secondary storage device to minimize average retrieval time, based on a knowledge of the probability for accessing the records. Theorems are presented for two limiting cases. A numerical example for an intermediate case is also given.

Programming by questionnaire: An effective way to use decision tables, D. W. Low, *Communications of the ACM* 16, No. 5, 282-286 (May 1973). Programming by questionnaire combines aspects of decision table programming by using decision tables to construct an application program through the selection of certain source statements from a predefined file. It is proposed that programming by questionnaire is a useful compromise between general and special purpose programming for a significant class of large scale problems. The elements of the approach are discussed and an existing application is described.

Register assignment algorithm for generation of highly optimized object code, J. C. Beatty, *IBM Journal of Research and Development* 18, No. 1, 20-39 (January 1974). A register assignment algorithm is described that, in contrast to traditional methods, permits a high level of optimization at both local and global levels. This involves splitting local register optimization into two phases, with global assignment intervening. Because novel techniques are used in the global assignment procedure, it is described in detail. Experimental results with a prototype implementation are presented in which object code improvements on the order of 25 percent over a production optimizing complier were obtained. No attempt was made to assess manpower cost of a final implementation nor to weight them against expected improvements in generated code.

Security, accuracy and privacy in computer systems, H. T. Martin, Prentice-Hall, Inc., Englewood Cliffs, New Jersey (1973). This book discusses the prevention of unauthorized access to computers and data banks, embezzlement, sabotage, and the invasion of privacy. Four layers of controls on such data are discussed. From the inner-most layer to the outer-most layer they are administrative controls, technical controls, physical controls, and societal and legal controls. The book offers a detailed explanation of techniques for achieving security and for controlling accuracy in computer systems. A checklist for designers and auditors is included at the end.