DATA STRUCTURES AND ALGORITHMS

Tasks and objectives

The course is intended for mastering the methods of displaying complex mathematical models that describe domain objects and operation with them on computer by means of software. The course includes basic data structures and their effective realization. It also provides many examples of use of the studied data structures for solution of complex tasks in different application domains using the principles of object-oriented programming (in C ++).

The course provides an introduction to development of parallel programs for systems with distributed memory and MPI.

Student Requirements

Knowledge of K101 "Introduction to Programming Methods 1", K102 "Introduction to Programming Methods 2", M207 "Discrete Mathematics", M210 "Algebra and Geometry". For performance of practical and laboratory tasks knowledge of C ++ programming methods is required.

Course description

The main sections of the course are:

- **Introduction – 2 academic hours**
  Importance of the subject. The problem of proving program correctness. Ways to decrease complexity of software.
- **Action Structure and Data Structure – 18 academic hours**
  The concepts of data structure and storage structure. Aggregate and special matrixes storage structures. Dynamic data structures: stack, queue. Memory management by repacking of storage structures. Structure of storage of several stacks in a common array of memory. Presentation of basic relationships using pointers. Linear list. Realization of lists with using dynamically distributed memory. Examples of use of stacks: radix sorting, conversion of arithmetic expressions to Polish notation.
- **Dynamic Structures and Displaying Complex Math Models on Computers – 22 academic hours**
  A system for arithmetic operations with multivariate polynomials. Presentation of polynomials using stacks and the problem of memory repacking. Presentation of polynomials in lists. Circular lists. Text editing. Presentation of text in the form of a hierarchical threaded list. Memory management in work with threaded lists, garbage collection. Structures of geometric entities' storage, plexes. Plex as a presentation of an arithmetic expression.
- **Name Accessing – 24 academic hours**
  Tabular data presentation, table concept, basic operations. Look-up and ordered tables, exhaustive and binary search. Presentation of tables in the form of search trees. Tables with computable addresses. Comparative analysis of table organization methods.
- **Introduction to Development of Parallel Programs for Systems with Distributed Memory ^{PAR} – 6 academic hours**
  Importance of the problematics of parallel calculations. Examples of state-of-the-art parallel computing systems. Principles of development of parallel programs: division of calculations and ensuring information independence. Indicators of quality of parallel calculations: acceleration and efficiency. Traits of development for distributed memory systems: data distribution and interaction based on message transmission. Introduction to MPI: processes, communicators, pair operations in data transmission. MPI programs setup using Microsoft Visual Studio framework. Start of MPI programs in local and distributed modes. Examples of parallel programs: matrix calculations, Monte-Carlo methods.

The course includes expanded laboratory training.

Course materials:

- electronic training materials;
- presentations for lectures;
- laboratory training sets.

Expected Training Results

Trainees who successfully completed the training course will acquire the following knowledge and skills:

- application of basic methods of mathematical structures presentation;
- realization of basic data structures: stacks, queues, threaded lists, trees, graphic charts, tables;
- realization of fundamental data processing algorithms: bypass, insertion, deletion of elements, sorting, search;
- estimation of complexity of basic data processing algorithms;
- application of object-oriented data storage structure development principle;
- initial level parallel programming using the MPI library.

References

1. Alfred V. Aho, Jeffrey D. Ullman, John E. Hopcroft Data Structures and Algorithms, Williams Publishing, 2000.
2. JanetPrichard, Frank M. Carrano Data Abstraction and Problem Solving: Walls and Mirrors (3rd Edition), Williams Publishing, 2003.
3. Donald E. Knuth Art of Computer Programming, Volume 1: Fundamental Algorithms (3rd Edition), Williams Publishing, 2000.
4. Gergel V.P. Theory and Practice of Parallel Calculations, Internet University of Information Technologies, Binom, 2007.

Website:

http://www.software.unn.ru/?doc=507

Contacts:

Gergel V.P., Dean of the Computing Mathematics Department (NNSU), Doctor of Science, gergel@unn.ru