Design and implementation of Univariate Polynomial Calculator

1.Statement

In scientific research, there are two different kinds of computational problems, (1) Numerical calculation (2) Symbolic computation. Symbolic computation is widely-used and challenging. The most representative and popular general symbolic computing software in the world are: MACSYMA，REDUCE，mu-MATH，MAPLE，Mathematica，Matlab，Python, etc. Real coefficient univariate polynomial operation is one of the most basic symbolic computing tasks. **In this experiment, students are required to use the basic principles of data structure, advanced computer programming language and related tools to analyze and solve the operation problems of univariate polynomial in the field of symbolic calculation, and realize the operations of addition, subtraction, multiplication, evaluation, derivation and integration of univariate polynomials.**

2. Experimental objectives

2.1 Be able to use appropriate logical structure to express univariate polynomials of any power, and to design algorithms of univariate polynomial operation based on logical structure;

2.2 Be able to select the appropriate storage structure to store sparse polynomials efficiently;

2.3 Be able to use programming language to implement the univariate polynomial operation;

2.4 Be able to design typical cases to test the correctness of algorithm implementation.

3. Experimental content

3.1 Analyze the requirements of univariate polynomial operation, design the logical structure of univariate polynomial, and define the ADT(Abstract Data Types), including algorithms of univariate polynomial operation;

3.2 To satisfy the needs of efficient storage and processing of univariate polynomials, research or design a variety of storage schemes, analyze its advantages and disadvantages, and give the conclusions;

3.3 Combined with the selected storage structure, algorithms of addition, subtraction, multiplication, evaluation and derivation of univariate polynomial is implemented;

3.4 Design experiments, randomly generate 10 groups of univariate polynomials of different scales (including the number of items), analyze the relationship between the time consumption of univariate polynomials operations and the scale of polynomials, and draw the chart of experimental results.

4. Experimental requirements

4.1 Functional requirements

(1)Randomly generate univariate polynomials of any power and any term number, write them to the file;

(2)Read the univariate polynomial from the file, specify two polynomials and complete various operations, and write the operation results to the file for inspection;

(3)Record the number of basic operations and measure the time consumption of univariate polynomial operations.

4.2 Preview requirements

1）Watch the experimental teaching video, complete the preview task, and write the preview report;

2）Learn the video, master the file processing technology, random number generation method, time measurement technology, etc;

3）Analyzes the problem of univariate polynomial, designs its logical structure, defines ADT and the algorithms, describes the pseudo code and flow chart of all algorithms;

4）Research or design a variety of storage schemes, analyze the advantages and disadvantages of different storage schemes in storage density, algorithm efficiency, implementation difficulty and other aspects, and form a conclusion.

5）High level programming language is used to implement the algorithms, the test data are designed and experiments are carried out to evaluate the correctness and efficiency of the algorithm.

4.3 Analysis and test requirements

1）A number of univariate polynomials with different lengths and shapes are generated randomly;

2）Read the univariate polynomial pairs which is randomly generated from the file, and specify them to complete a certain kind of operation, and write the operation process to the file for checking;

3）From the file, the randomly generated polynomial pairs of different sizes are read in. The operation scale and time efficiency are measured and recorded. The experimental results chart is drawn to obtain effective conclusions.

4.4 Requirements for submission of results

1) Write a preview report;

2) Write one experiment report;

3) Submit the electronic documents of experiment preview report, experiment report, program code, test data and test results on the course platform. All files should be in .docx format, and the file name format should be Class-Name-Experiment 1-Task name.docx, such as "ComputerUC191-Zhang San-Experiment 1-Preview report.docx".

5.Expanding tasks

1. How to realize the division of univariate polynomial?

2. How to determine whether a univariate polynomial can be factorized and how to implement it?

3. How to solve the problem of sign operation of binary polynomials?

4. If univariate polynomials is stored by array, how to implement the multiplication, and how about its efficient?

5. The index of the accumulated items generated in the multiplication operation is usually piecewise increasing. Can the multiplication algorithm be optimized by reducing the time consumption of inserting the add item?

6. How to deal with the univariate polynomial with nonstandard format?