

Задание L1B

Библиотеки функций

В следующих задачах требуется написать библиотеку функций с заданным интерфейсом для работы с определённым типом данных и программу, демонстрирующую возможности библиотеки. Таким образом, решение должно состоять из двух проектов:

1. Библиотека `xxx`: создаётся `xxx.lib`, которая собирается из `xxx.c` и `xxx.h`.
2. Консольное приложение `xxxtest`: создаётся `xxxtest.exe`, которое собирается из `xxxtest.c` с использованием `xxx.h` и `xxx.lib`.

Исходный текст библиотеки (заголовочный файл) должен содержать комментарии для каждого экспортируемого объекта (функции, типа, константы), для автоматической генерации документации в формате HTML/CHM с помощью утилиты Doxygen.

Все функции библиотеки традиционно в качестве префикса имён должны иметь название типа, чтобы отличать от аналогичных функций для другого типа, например, для типа `xxx_t` функции `XxxCreate()`, `XxxAdd()` и т.п.

Комментарии к задаче

Интерфейс Краткое описание возможной функциональности для различного вида чисел или аналогичных объектов:

- `XxxCreate`, `XxxDestroy`: создание по набору параметров, с выделением динамической памяти при необходимости, уничтожение, в первую очередь динамической памяти;
- `XxxFromYyy`, `XxxAsYyy`: создание из определенного типа, преобразование к определенному типу;
- `XxxParse`, `XxxFormat`: создание по строке, запись в строковый буфер;

- XxxRead, XxxWrite: создание из потока, запись в поток, в текстовом виде;
- XxxAdd, XxxSub, XxxMul, XxxDiv: сложение, вычитание, умножение, деление;
- XxxNegate, XxxReciprocal, XxxInverse: противоположное, обратное;
- XxxCompare: сравнение (функция сравнения традиционно принимает два указателя и возвращает -1 , 0 или $+1$ когда первый аргумент меньше, равен или больше второго соответственно);

План решения

Предлагаем вам следующие шаги решения.

1. Объявите...

Варианты

Вариант LIB-1 (Комплексные числа). В рамках общего условия задачи ввести новый прозрачный тип «комплексное число» (`complex_t`) на основе значений `re`, `im` типа `double` и реализовать функциональность:

- константа `ZERO`;
- `Create`;
- `Parse`, `Format`, `Read`, `Write`, в формате `"(1, -0.5)"`;
- `Add`, `Sub`, `Mul`, `Div`, `Negate`, `Reciprocal`;
- `Abs`, `Arg`: модуль, аргумент;
- `Conjugate`: сопряжение;

Вариант LIB-2 (Нечёткая логика). В рамках общего условия задачи ввести новый прозрачный тип «fuzzy bool» (`fbool_t`) на основе значения типа `double` и реализовать функциональность:

- `FromInt`, `FromDouble`, `AsInt`, `AsDouble`;
- `Parse`, `Format`, `Read`, `Write`, в формате "`fbool[0.25]`";
- `And`, `Or`, `Not`: конъюнкция, дизъюнкция, отрицание;
- `Compare`;

Вариант LIB-3 (Рациональные дроби). В рамках общего условия задачи ввести новый прозрачный тип «несократимая рациональная дробь» (`rational_t`) на основе значений `num`, `den` типа `int` и реализовать функциональность (не забывая сокращать дробь):

- константа `ZERO`;
- `Create`; `FromDouble`^(*), `AsDouble`, `AsInt`;
- `Parse`, `Format`, `Read`, `Write`, в формате "`-2\3`";
- `Add`, `Sub`, `Mul`, `Div`, `Negate`, `Reciprocal`, `Abs`;
- `Round`, `Floor`, `Ceil`: округления; `Compare`;

Вариант LIB-4 (Вещественные числа с фиксированной точкой). В рамках общего условия задачи ввести новый прозрачный тип «с фиксированной точкой» (`fixed_t`) на основе типа `int` (напр., 32 бита как 16:16) и реализовать функциональность:

- константа `ZERO`;
- `FromInt`, `FromDouble`, `AsInt`, `AsDouble`;
- `Parse`, `Format`, `Read`, `Write`, в формате "`Fix[-12.75]`";
- `Add`, `Sub`, `Mul`, `Div`, `Negate`, `Reciprocal`, `Abs`;
- `Round`, `Floor`, `Ceil`: округления; `Compare`;

Вариант LIB-5 (Длинные целые числа). В рамках общего условия задачи ввести новый прозрачный 18-значный тип «очень длинный» (`verylong_t`) на основе двух значений `hi`, `lo` типа `long` (в каждом хранить по 9 десятичных цифр, в старшем числе знак) и реализовать функциональность:

- константа `ZERO`;
- `FromLong`, `AsLong`;
- `Parse`, `Format`, `Read`, `Write`;
- `Add`, `Sub`, `Mul`, `Div`, `Negate`, `Abs`; `Compare`;

Вариант LIB-6 (Приближённые вещественные числа). В рамках общего условия задачи ввести новый прозрачный тип «приближённое число» (`approx_t`) на основе двух значений `lo`, `hi` типа `double` и реализовать функциональность:

- `FromDouble`: по центру и радиусу; `FromRange`, `AsDouble`, `GetRadius`;
- `Parse`, `Format`, `Read`, `Write`, в формате "`<0.75; 1.25>`";
- `Add`, `Sub`, `Mul`, `Div`, `Negate`, `Abs`; `Compare`;

Вариант LIB-7 (Матрицы 2×2). В рамках общего условия задачи ввести новый прозрачный тип «матрица 2×2 » (`matrix22_t`) на основе значений `a`, `b`, `c`, `d` типа `double` и реализовать функциональность:

- константы `ZERO`, `IDENTITY`;
- `SetZero`, `SetIdentity`: заполнение как нулевой и единичной;
- `Parse`, `Format`, `Read`, `Write`,
в формате "`{{1, 0}, {-3.5, 1}}`";
- `Add`, `Sub`, `Mul`, `MulDouble`;
- `Determinant`, `Trace`, `Inverse`, `Transpose`;
- `Element`: доступ к элементу A_{ij} ;

Вариант LIB-8 (Матрицы произвольного размера). В рамках общего условия задачи ввести новый непрозрачный тип «матрица» (`matrix_t`) на основе значений типа `double` и реализовать функциональность:

- Create, Destroy: в динамической памяти;
- SetZero, SetIdentity: заполнение как нулевой и единичной;
- Parse, Format, Read, Write, в формате "{1, 0}, {-3.5, 1}";
- Add, Sub, Mul, MulDouble;
- $Determinant^{(*)}$, Trace, $Inverse^{(*)}$, Transpose;
- Element: доступ к элементу A_{ij} ;

Примечание: в качестве типа рекомендуется использовать структуру с тремя полями, хранящими число строк, столбцов и указатель на линейную память под элементы.

Вариант LIB-9 (Многочлены). В рамках общего условия задачи ввести новый непрозрачный тип «многочлен» (`poly_t`) на основе значений типа `int` и реализовать функциональность:

- Create, Destroy: по массиву, в динамической памяти;
- From: по N коэффициентам, с переменным числом аргументов;
- Parse, Format, Read, Write, в формате " $2x^3 - 5x + 7$ ";
- Add, Sub, Mul, MulInt; Evaluate: вычислить в точке;
- Element: доступ к элементу A_i ;

Примечание: в качестве типа можно использовать структуру с двумя полями, хранящими степень многочлена и указатель на выделенную память под коэффициенты.

Вариант LIB-10 (Векторы в 2D). В рамках общего условия задачи ввести новый прозрачный тип «вектор на плоскости» (`vec2_t`) на основе значений `x`, `y` типа `double` и реализовать функциональность:

- константы `ZERO`, `UNIT_X`, `UNIT_Y`;
- `Parse`, `Format`, `Read`, `Write`, в формате "{1.5, -3.5}";
- `Add`, `Sub`, `MulDouble`, `Negate`, `Length`, `Normalize`;
- `Dot`, `Cross`: скалярное и векторное умножение;
- `TurnLeft`, `TurnRight`, `Rotate`: повороты на 90° и на произвольный угол;

Вариант LIB-11 (Строки). В рамках общего условия задачи ввести новый непрозрачный тип «динамическая строка» (`string_t`) и реализовать функциональность:

- `Create`, `Destroy`: по Си-строке, в динамической памяти;
- `Parse`, `ParseWord`, `Read`, `ReadWord`, `Write`: отдельно чтение до перевода строки и до пробельного символа;
- `Length`, `Copy`, `Append`, `Compare`,...;
- `Element`: доступ к элементу S_i ;

Вариант LIB-12 (Вещественные числа с десятичной точкой). В рамках общего условия задачи ввести новый прозрачный тип «с фиксированной точкой», но не в двоичной, а в десятичной форме (`fixed10_t`), на основе типа `int` с двумя десятичными знаками после точки и реализовать функциональность:

- константа `ZERO`;
- `FromInt`, `FromDouble`, `AsInt`, `AsDouble`;
- `Parse`, `Format`, `Read`, `Write`, в обычном формате "Dec[-12.75]";
- `Add`, `Sub`, `Mul`, `Div`, `Negate`, `Reciprocal`, `Abs`;
- `Round`, `Floor`, `Ceil`: округления; `Compare`;