

Раздел «Язык Си» . ComplexLib :

Создание библиотек. Библиотека для работы с комплексными числами.

- [Создание библиотеки](#)
- [Пример использования библиотеки](#)
- [Создание и использование библиотек в Gnu C](#)
 - [Подключение библиотеки в виде исходников](#)
 - [Подключение библиотеки в виде статически загружаемой библиотеки](#)
 - [Задачи](#)

Создание библиотеки

Библиотека у нас будет иметь два исходных файла: complex.h и complex.cpp

```
/*
  complex.h
*/
typedef struct {
    double a,b;
} complex_t;

complex_t mul(complex_t x, complex_t y);
```

```
/*
  complex.c
*/
#include "complex.h"
complex_t mul(complex_t x, complex_t y)
{
    complex t;
    t.a = x.a * y.a - x.b * y.b;
    t.b = x.a * y.b + x.b * y.a;
    return t;
}
```

Пример использования библиотеки

Теперь несложно использовать эту библиотеку. Для этого нужно:

- добавить в заголовок c/cpp-файла строчку `#include` ;
- сделать так, чтобы компилятор находил `complex.h` ;
- сделать так, чтобы сборщик (linker) находил файл библиотеки `complex` .

```
/*
  test_complex.c
*/
#include <stdio.h>
#include <complex.h>

int main()
{
    complex_t x = {1,2};
    complex_t y = {3,4};
    complex_t z = mul(x, y);
    printf ("z = (%lf, %lf)\n", z.a, z.b );
    return 0;
}
```

Создание и использование библиотек в Gnu C

Подключение библиотеки в виде исходников

Есть простой способ подключения библиотеки функций прямо в виде исходников:

```
$ gcc test_complex.c complex.c -o test_complex
```

Поиск

 Поиск

Раздел «Язык Си»

[Главная](#)
[Зачем учить C?](#)
[Определения](#)

Инструменты:

[Поиск](#)
[Изменения](#)
[Index](#)
[Статистика](#)

Разделы

[Информация](#)
[Алгоритмы](#)
[Язык Си](#)
[Язык Ruby](#)
[Язык Ассемблера](#)
[E! Judge](#)
[Парадигмы](#)
[Образование](#)
[Сети](#)
[Objective C](#)

Login>>

В итоге получается запускаемый файл `test_complex`.

Подключение библиотеки в виде динамически загружаемой библиотеки

```
$ gcc -shared complex.c -o libcomplex.so           # создание динамически загружаемой библиотеки
$ gcc test_complex.c -L. -lcomplex -o test_complex # компиляция тестовой программы
$ export LDD_LIBRARY_PATH=. ; ./test_complex      # запуск тестовой программы
```

- Есть ряд стандартных путей (`LD_LIBRARY_PATH`), где система ищет динамически загружаемые библиотеки. Текущая директория по умолчанию не является таким местом, поэтому мы явно указываем, что библиотеки могут лежать также и в текущей директории. Переменная `LDD_LIBRARY_PATH` содержит дополнительный список путей (разделённых двоеточием), где система будет искать библиотеки.
- Расширение `.so` соответствует `shared objects`, то есть библиотека объектов общего доступа.

Подключение библиотеки в виде статически загружаемой библиотеки

Для компилятора `glibc` команды создания статически подключаемой библиотеки и подключения её к проекту выглядят следующим образом

```
$ gcc -c complex.c -o libcomplex.a               # создание динамически подключаемой библиотеки
$ gcc test_complex.c -L. -lcomplex -o test_complex # компиляция тестовой программы
$ ./test_complex                                # запуск тестовой программы
```

Задачи

- Напишите расширенную версию библиотеки функций, для работы с комплексными числами. Реализуйте операции сложения, вычитания, умножения и деления. А также функции, вычисляющие аргумент и модуль комплексного числа. Реализуйте функции комплексного аргумента `sinc`, `cosc`, `exp`, `log`, `asinc`, `acosc`. Чему равен арккосинус двух (`acosc(2) = ?`) ? Напишите программу, тестирующую вашу библиотеку.
- Напишите библиотеку функций по теме "аналитическая геометрия"
 - Решать задачи на плоскости
 - вычислять расстояние для пар (точка, точка), (точка, прямая)
 - находить площадь многоугольника
 - проверять, пересекаются ли две прямые и находить точку их пересечения
 - проверять, пересекаются ли два отрезка и находить точку их пересечения
 - Решать задачи в пространстве
 - вычислять расстояние для пар (точка, точка), (точка, прямая), (прямая, прямая), (точка, плоскость)
 - находить точку пересечения трёх попарно непараллельных плоскостей
 - находить объём тетраэдра

-- ArtemVoroztsov - 17 Nov 2004

(с) Материалы раздела "Язык Си" публикуются под лицензией GNU Free Documentation License.