

Поиск:

Вперед

Ссылка <cstdarg>

Зарегистрирован как: legioner9

Учетная запись

Выход

C++
Информация
Руководства
Ссылка
Статьи
Форум

Ссылка
Библиотека C:
<cassert> (assert.h)
<cctype> (ctype.h)
<cerrno> (errno.h)
<cfenv> (fenv.h)
<cfloat> (float.h)
<cinttypes> (inttypes.h)
<ciso646> (iso646.h)
<climits> (limits.h)
<locale> (locale.h)
<cmath> (math.h)
<setjmp> (setjmp.h)
<signal> (signal.h)
<cstdarg> (stdarg.h)
<stdbool> (stdbool.h)
<stddef> (stddef.h)
<stdint> (stdint.h)
<stdio> (stdio.h)
<stdlib> (stdlib.h)
<string> (string.h)
<tgmath> (tgmath.h)
<time> (time.h)
<uchar> (uchar.h)
<wchar> (wchar.h)
<wctype> (wctype.h)
Контейнеры:
Ввод/вывод:
Многопоточность:
Другое:

<cstdarg> (stdarg.h)
макросы:
va_arg
va_copy
va_end
va_start
типы:
va_list

заголовок

## <cstdarg> (stdarg.h)

### Обработка переменных аргументов

Этот заголовок определяет макросы для доступа к отдельным аргументам списка неназванных аргументов, число и типы которых не известны вызываемой функции.

Функция может принимать различное количество дополнительных аргументов без соответствующих объявлений параметров, включая запятую и три точки (,...) после ее обычных именованных параметров:

return\_type function\_name ( parameter\_declarations,...);  
для доступа к этим дополнительным аргументам можно использовать макросы `va_start`, `va_arg` и `va_end`, объявленные в этом заголовке:

- Во-первых, `va_start` инициализирует список аргументов переменных как `va_list`.
- Последующие выполнения `va_arg` дают значения дополнительных аргументов в том же порядке, что и переданные функции.
- Наконец, `va_end` должен быть выполнен до возврата функции.

### Типы

va_list	Тип для хранения информации об аргументах переменной (тип )
---------	---

### Макрофункции

va_start	Инициализация списка аргументов переменной (макроса )
va_arg	Получить следующий аргумент (макрос )
va_end	Конец списка аргументов переменной (макрос )
va_copy	Скопировать список аргументов переменной (макрос )