

Search: Go

Not logged in


Reference <ctime> clock

register

log in

C++
Information
Documentation
Reference
Articles
Forum

Mettere online un sito WordPress è questione di attimi.
Prova e capirai. **PROVALO GRATIS* PER 90 GIORNI**

 Windows Azure
* esclusi costi di connessione.

Reference

C library:

<cassert> (assert.h)
<cctype> (ctype.h)
<cerrno> (errno.h)
<cfloat> (float.h)
<ciso646> (iso646.h)
<climits> (limits.h)
<locale> (locale.h)
<cmath> (math.h)
<csetjmp> (setjmp.h)
<csignal> (signal.h)
<csdarg> (stdarg.h)
<csdbool> (stdbool.h)
<csddef> (stddef.h)
<csdint> (stdint.h)
<csdio> (stdio.h)
<csdlib> (stdlib.h)
<cstring> (string.h)
<ctime> (time.h)
<cuchar> (uchar.h)
<wchar> (wchar.h)
<cwctype> (wctype.h)

Containers:

Input/Output:

Other:

function

clock

<ctime>

```
clock_t clock (void);
```

Clock program

Returns the processor time consumed by the program.

The value returned is expressed in *clock ticks*, which are units of time of a constant but system-specific length (with a relation of `CLOCKS_PER_SEC` *clock ticks* per second).

The epoch used as reference by `clock` varies between systems, but it is related to the program execution (generally its launch). To calculate the actual processing time of a program, the value returned by `clock` shall be compared to a value returned by a previous call to the same function.

Parameters

none

Return Value

The number of clock ticks elapsed since an epoch related to the particular program execution.

On failure, the function returns a value of `-1`.`clock_t` is a type defined in `<ctime>` as an alias of a fundamental arithmetic type.

Example

```
1 /* clock example: frequency of primes */
2 #include <stdio.h>          /* printf */
3 #include <time.h>           /* clock_t, clock, CLOCKS_PER_SEC */
4 #include <math.h>           /* sqrt */
5
6 int frequency_of_primes (int n) {
7     int i,j;
8     int freq=n-1;
9     for (i=2; i<=n; ++i) for (j=sqrt(i);j>1;--j) if (i%j==0) {--freq; break;}
10    return freq;
11 }
12
13 int main ()
14 {
15     clock_t t;
16     int f;
17     t = clock();
18     printf ("Calculating...\n");
19     f = frequency_of_primes (99999);
20     printf ("The number of primes lower than 100,000 is: %d\n",f);
21     t = clock() - t;
22     printf ("It took me %d clicks (%f seconds).\n",t,((float)t)/CLOCKS_PER_SEC);
23     return 0;
24 }
```

Output:

```
Calculating...
The number of primes lower than 100,000 is: 9592
It took me 143 clicks (0.143000 seconds).
```

Data races

Concurrently calling this function is safe, causing no data races.

Exceptions (C++)

No-throw guarantee: this function never throws exceptions.

See also

time	Get current time (function)
difftime	Return difference between two times (function)