

C/C++中 size_t 与C#中类似的类型定义_c++ size_t 对应c#_丕兜兜飞己的博客-CSDN博客

成就一亿技术人!

在c/c++中, 关于size_t, 通常情况下, 32位程序中就是unsigned int,64位中就是unsigned long long。

但是如果函数跨语言封装中用到了size_t则是一个不怎么美好的事情了, 比如C#中并没有对应的可变类型定义, 当然你可以直接用using指明使用那种类型代替size_t,如:

```
#if WIN32
    using size_t = System.UInt32;
#else
    using size_t = System.UInt64;
#endif
```

1
2
3
4
5
6
7

可以看出这并不那么美观..., 而且还得定义条件编译符号。所以, 有些开源项目中的size_t使用了System.UIntPtr来代替, 那么System.UIntPtr有什么特殊之处呢, 查看.net frame work的源码<http://referencesource.microsoft.com/#mscorlib/system/uintptr.cs,f8f01749ea279a65>

可以看到, 实际上也是使用了条件编译, 但用起来相对简单些不是吗

```
using size_t = System.UIntPtr;
```

1
2

感兴趣的也可以看下这两篇的讨论

https://bytes.com/topic/c-sharp/answers/458667-interop-size_t

<https://stackoverflow.com/questions/12966283/how-to-marshall-size-t-cross-platform-using-semantic-types>