

编译原理第十次理论作业

18340052 何泽

Surf the Internet and write a short paper to compare GC on Java platform with GC on Microsoft .Net platform.

二者均使用可达性分析法来查找没有被引用的对象，但是二者的GCRoots不同，

Java:

- 虚拟机栈（栈帧中的局部变量区，也叫做局部变量表）中引用的对象
- 方法区中的类静态属性引用的对象
- 方法区中常量引用的对象
- 本地方法栈中JNI(Native方法)引用的对象

.NET:

- 全局对象和静态对象的引用
- 应用程序代码库中局部对象的引用
- 传递进一个方法的对象参数的引用
- 等待被终结对象的引用
- 任何引用对象的CPU寄存器

除此之外，二者在分代垃圾回收法的实现细节不同：

.NET被分配的对象划分为3个类别，分别为0、1、2，最近被分配内存空间的对象被放置于第0代，因为第0代很小，小到足以放进处理器的二级（L2）缓存，所以第0代能够为我们提供对其中对象的快速存取；经过一轮垃圾回收后，仍然保留在第0代中的对象被移进第1代中；再经过一轮垃圾内存回收后，仍然保留在第1代中的对象则被移进第2代中。第2代包含了生存期较长的对象，这些对象至少经过了两回收。垃圾回收器会先检查所有的0代对象，如果还需要更多的空间，那么垃圾回收器会继续检查所有的1代对象，整理出足够的空间。这时，没有被回收的1代对象将成为2代对象。分代可以避免每次垃圾回收都遍历整个托管堆，这样可以提高垃圾回收的性能。

而Java使用分代收集器，它把堆分为三个主要的域：新域、旧域以及永久域。JVM生成的所有新对象放在新域中；一旦对象经历了一定数量的垃圾收集循环后，便获得使用期并进入旧域。