Alan Moody

Garbage Collection

Mark & Sweep and Large Objects

Mark and Sweep is an algorithm for garbage collection that values space over time. So, when working with a large object space, this can be the preferred approach. Two common disadvantages to Mark and Sweep are: 1) It will fragment memory, making the process less efficient. And 2) It’s running time is proportional to the size of the heap, rather than the amount of live memory. In the case of a large object space however, it seems that the garbage collection time is generally better than allocation time, and so the mark and sweep would be the better choice. The disadvantages to mark and sweep are much less sever for large objects as they are for small ones. The cost of allocation in mark and sweep is a ‘per object’ cost. Fragmentation is also easier to avoid with large objects.

Type Tag Placement

It shouldn’t matter where the tag is placed. Whether it is at the beginning or the end, we will have to examine the whole thing anyway.

Intergenerational Pointers

Generational garbage collectors seem to be sort of the middle ground between stop and copy and mark and sweep. In this case, memory is divided into two areas: one that is collected using stop and copy, and the other is collected using mark and sweep. The advantage is that you can preserve long term objects longer, while still deallocating short term objects when you don’t need them anymore. Disadvantages include managing the references that cross over between the two areas.

Application

Understanding garbage collection will be useful in most applications, though it isn’t of too much concern in garbage collected languages like Java. When programming in C++ however, it is essential to know the ins and outs of how your memory is being managed in order to have a time and space efficient program.

In general, garbage collection is an important part of writing well thought out code. I will be able to use this knowledge to control my memory allocation and deallocation, and overall have more control over the things I create.