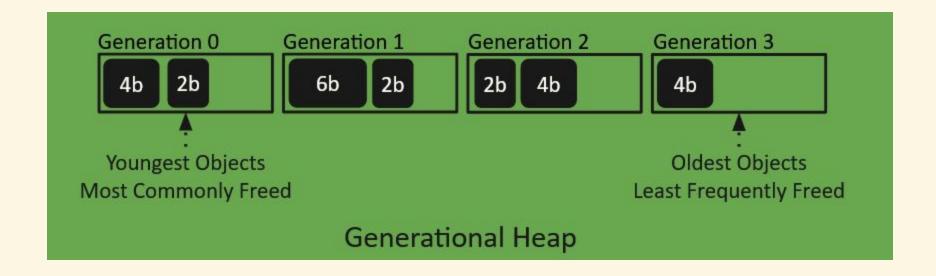
Heap Heap Hooray

Milestone 6

Finished Generational GC

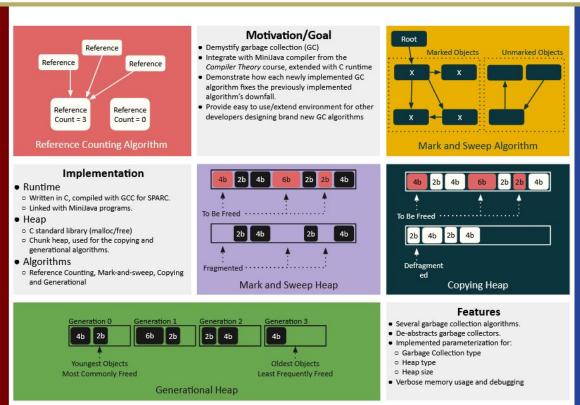




Heap Heap Hooray: Reinventing the Garbage Collector

Trevor Schiff, Tyler Gutowski

Faculty Advisor(s): Ryan Stansifer, Dept. of Electrical Engineering and Computer Science, Florida Institute of Technology



Software Engineering

Project Name Heap Heap Hooray

Team Lead: Trevor Schiff

Team Member(s): Trevor Schiff, Tyler Gutowski

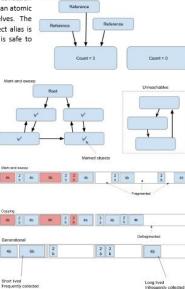
Faculty Advisor(s): Dr. Ryan Stansifer, Dept. Of Electrical Engineering and Computer Science

Heap Heap Hooray is an advanced, yet easy to understand memory management system for the MiniJava compiler, encompassing reference counting, mark-sweep, copying, and generational garbage collection techniques. The system aims to optimize memory usage, minimize fragmentation, enhance performance, and be fully customizable with parameters.

Reference counting is a simple garbage collection method that relies on objects maintaining an atomic count of existing references to themselves. The reference count is updated when an object alias is created or leaves scope, and the object is safe to free when this count reaches zero.

Mark-and-sweep is a garbage collection algorithm that finds and marks reachable objects via the program stack. Any items that have not been marked are unreachable (garbage), and can be freed. The algorithm sweeps through the heap, freeing anything that hasn't been marked. The issue with this algorithm is that the memory becomes fragmented, which can detract from performance.

Copying is a garbage collection algorithm that acts similarly to mark-and-sweep. Copying solves the fragmentation issue by using two heaps. During the sweep phase, everything that is marked is moved to the opposite heap, defragmenting the layout of these objects in the process. Lastly, unmarked objects are freed, emptying the first heap.



Generational is a garbage collection algorithm built on the copying algorithm, but with more heaps. Objects are categorized into different "generations" based on their age. The garbage collection efforts primarily target the younger generations, where short-lived objects are more prevalent, reducing the frequency and duration of garbage collection pauses.

Table of Contents		/h	40
System Design	03	gc.c/.h	10
Introduction	03	refcount_gc.c/.h	11
Runtime	03	marksweep_gc.c/.h	12
Garbage Collector	04	copying_gc.c/.h	12
Неар	04	generational_gc.c/.h	13
Containerization	05	Неар	13
Algorithms	05	heap.c./h	13
Reference Counting	06	chunk_heap.c./h	15
Mark-and-sweep	06	stl_heap.c/.h	15
Copying	07	Utilities	16
Generational	8	config.c./h	16
Testing and Evaluation	08	debug.c./h	10
Developer's Guide	09		
Installation	09	linklist.c./h	17
Jabberwocky	09	runtime.c./h	19
SPARC Container	10	stackframe.c./h	19
Garbage Collector	10	MJC Usage	20

```
void refcount_ref_decr(GC* gc, Object* obj);
  Decrements an object's reference count. If the counter reaches zero, the
  object is freed.
```

marksweep gc.c/.h

This file contains the mark-sweep garbage collector implementation. If you would like to use these functions, please use the following public API in marksweep.h:

```
GC* marksweep_create(void);
Create a mark-sweep garbage collector.

void marksweep_destroy(GC* gC);
Destroy this garbage collector.

void marksweep_collect(GC* gC);
Perform mark-and-sweep garbage collection cycle.

void marksweep_stack_push(GC* gc, void* frame, u32 size);
Pushes a new stack frame to traverse later using garbage collection.

void marksweep_pop_stack(GC* gC);
Pops the current stack frame.
```

copying gc.c/.h

This file contains the copying garbage collector implementation. If you would like to use these functions, please use the following public API in copying.h:

```
GC* copying_create(void);
Create a copying garbage collector.

void copying_destroy(GC* gC);
Destroy this garbage collector.

void copying_collect(GC* gC);
Perform a copying garbage collection cycle.

void copying_stack_push(GC* gC, void* frame, u32 size);
Push a new active stack frame to traverse later during garbage collection.
```

Jabberwocky

Install Poetry from python-poetry.org.

Install Jabberwocky from https://github.com/Kippiii/jabberwocky-container-manager.

Navigate to the installed Jabberwocky folder.

```
poetry install
poetry shell
python build.py
```

MJC Usage

```
Usage: {compiler-jar} [option(s)] input-file
```

Options:

--help Display this information again.

--verbose Log verbose compiler information to "/verbose.txt".

--gc=<type> Set <type> as the program's garbage collection method.

(None | RefCount | MarkSweep | Copying | Generational)

--heap=<type> Set <type> as the program's heap type.

(St1|Chunk|Buddy)

```
certisa bean bean at 80007 aller second $62008 (circultion compare $62018)
martine/head/head of MMALT the Willio (1654) EDIT
ruriles has here in 8000 alles asserts dubble balanciib, unaretri dubble
Portion/Reso, Page 1: 80007 aller amoses %:19000 faller.2110, casestro %:20000
runting hass hase in 88662 mg aller (step 1967)
 CONTERNATIONAL PRINCIPLE WITCH CARRY MILLION COMMONDS TANDAIN, PROVIDING AND CARRY
runtim interioframe middell valents stant Arens - middelesses Calles (60)
PART SHALL STREET THESE STREET, STREET
                                                             treat adinagin leasenguere
                                                             shir-15map(2)+090940408
PARTEROLOGICAL CORNER.
                                                             CONTRACTOR ASSESSMENT
POPULAR PERSONNEL LIBERT
rorting intuctifront, militari
                                                             APPLICATION OF A STREET
 roftles/Italian Free, EXIDE
                                                             ACCUPATION AND ADDRESS OF
rortino, at solv from a sufficient
                                                             THE PROPERTY OF STREET
rorting/Itach frame, Edges
                                                             1753 - 167546 F HOMESSON
                                                             もあるといのの中央に通りの問題の自然に対
riorithms (all mote France, p. 86467).
runtissitaturitryss.comitis: transcis c.shirisurets oc.aminissistic
POPELANA NEW YEAR ROUGHER SELECTED STOP AND A SHARE SHAPE BY 2002 BY
nortimalated from mility training alleged block because
 Purtine (Italy Frame, E. 9190). Transfra. 6. (6):+6:29009. 66.40(+6:29008
runtime textension per militally marketines mark to 2006.
reintlematerials from a [1,8185] transfer allered block excesses
runtime. (Atlantiframer, p. 81827) Arestorne ar abit-discretion on abit-final con-
riortimology/narksonap.go. 0.5685] marksonap staric bulleton
rurlles/atmosfrues, n.H.1807 transpase alleades $15a8 4x28548
runtiamietum francis (1818) francesa a phintelialist an anti-militari
PORTLANDAR/MARKINGHALAS; CORROL HARRISANDE RAPA INCRESA
runtima intucirfrom, milital i trainrea laifeirai biscir desitibli
PARTERNATURESTYMBER TURBERT TYMBERER ILLERSHEISTERN DISJESSFREISTERN
Purties (ga/markousep ga. c. $180) marketens work $125568
rorting at word from a distant transfer all most black departm
                                                             $753 - 0 Gallet $1573 + $660000000
                                                             state of George [2] + $680000000
PARTINGSTERNATIONS, CHEET)
                                                             123-100440 11-10000
runsiam factorie France, unit $873
                                                             tive - I diswayed it I with a position
PROPERTY AND ADDRESS OF THE PARTY.
                                                             RECEIVED AND ADDRESS OF
parties, attach from the Hills (
                                                             Colored, Income?
Purties [hasp hase in METO] cant aller semesses !!!
 经收益的
```

Executing a fragmentation test using the markand-sweep GC.

Although enough memory exists overall, it is not contiguous. Fragmentation results in the program terminating.