Can we avoid sweep phase?
Can we design an $O(A)$ garbage collection algorithm.
Move all accessible cells somewhere else & what remains is garbage.
Copying collection

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>B</td>
<td>C</td>
</tr>
<tr>
<td>1</td>
<td>D</td>
<td>E</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>G</td>
</tr>
<tr>
<td>3</td>
<td>H</td>
<td>I</td>
</tr>
<tr>
<td>4</td>
<td>J</td>
<td>K</td>
</tr>
</tbody>
</table>

Use in-place DFS

divide memory in half
Phase 1: Move reachable cells to other half

"From" in use

"To" other half

+ leave forwarding address

Phase 2: Update all reference in to half using forwarding address

Then switch to/from
Time Complexity

\[
O(A)
\]

For copying collection

Cost of garbage collection
\[
\frac{\text{# cells freed}}{A}
\]

Mark+ Sweep
\[
O\left(\frac{A+M}{M-A}\right)
\]