**Overview**

- ASMprofiler provides a tool that allows for efficient analysis of programs written in Assembly language.
- It is designed to work with both x86 and AMD64 architectures.
- The tool provides detailed information about the execution of programs, including call graphs and data access patterns.

**Motivation**

- The tool was developed to address the need for a comprehensive profiling tool for Assembly language programs.
- It aims to provide insights into program behavior that are not easily accessible through other profiling tools.
- The tool is designed to be user-friendly and accessible to both experts and beginners in Assembly language.

**Python3 Script Options**

- `-domain DOMAIN` - This option is added to show a domain name that is not local to the script.
- `-print PRELOAD` - This option is added to print the loaded library with the program compiled without our library.
- `-show_history SHOW_HISTORY` - This option is added to show the history of the program.
- `-show_address SHOW_ADDRESS` - This option is added to show the address of the program.
- `-show_code SHOW_CODE` - This option is added to show the code of the program.

**Display Data (Grouping by Function Block)**

<table>
<thead>
<tr>
<th>Function</th>
<th>Time</th>
<th>Calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function1</td>
<td>10.5</td>
<td>200</td>
</tr>
<tr>
<td>Function2</td>
<td>7.2</td>
<td>150</td>
</tr>
<tr>
<td>Function3</td>
<td>9.8</td>
<td>300</td>
</tr>
</tbody>
</table>

**Example Usage**

- First, the program runs with the profiler and the list of top time taking instructions is analyzed.

**Conclusion**

- ASMprofiler is useful for optimizing assembly and C programs.
- It provides detailed insights into program behavior, making it a valuable tool for developers and researchers.

**Display Data (Grouping by C Source Line Functionality)**

<table>
<thead>
<tr>
<th>Source Line</th>
<th>Function</th>
<th>Time</th>
<th>Calls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line1</td>
<td>Function1</td>
<td>8.7</td>
<td>100</td>
</tr>
<tr>
<td>Line2</td>
<td>Function2</td>
<td>6.3</td>
<td>150</td>
</tr>
<tr>
<td>Line3</td>
<td>Function3</td>
<td>9.2</td>
<td>250</td>
</tr>
</tbody>
</table>

**Example Usage**

- 1000 runs of the optimized and unoptimized version were run and the average was taken.

- Unoptimized: 4.282475
- Optimized: 4.08466