**GROMACS - Bug #716**

**Memory leak in g_rms when using the -prev flag**

03/02/2011 05:53 PM - Daniel Larsson

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Low</td>
</tr>
<tr>
<td>Assignee:</td>
<td>David van der Spoel</td>
</tr>
<tr>
<td>Category:</td>
<td></td>
</tr>
<tr>
<td>Target version:</td>
<td>future</td>
</tr>
<tr>
<td>Affected version - extra info:</td>
<td>4.5.3 and earlier</td>
</tr>
<tr>
<td>Affected version:</td>
<td>5.0</td>
</tr>
<tr>
<td>Difficulty:</td>
<td>uncategorized</td>
</tr>
</tbody>
</table>

**Description**

g_rms eats up all system memory and crashes when analyzing long and large trajectories using the -prev flag.

**Associated revisions**

Revision d40ce2b8 - 06/09/2014 09:42 PM - Rossen Apostolov

Issue a warning for using gmx_rms -prev with large trajectories.

Refs #716.

Change-id: I607d6a63259e0dd9de7c8c99e4917bbe9f37f8e

**History**

#1 - 04/11/2012 12:13 PM - Rossen Apostolov

- Assignee set to David van der Spoel
- Priority changed from Normal to High
- Target version set to 4.5.6

#2 - 04/12/2012 09:07 PM - David van der Spoel

- Priority changed from High to Low
- Target version changed from 4.5.6 to 5.0

To fix this would require a large amount of work. This can be postponed to a later release when we do more modularization.

#3 - 05/23/2014 01:31 AM - Erik Lindahl

- Affected version set to 5.0

Rossen: I would consider removing the -prev feature for now.

#4 - 05/23/2014 01:49 PM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #716.
Uploader: Rossen Apostolov (rossen@kth.se)
Change-id: I607d6a63259e0dd9de7c8c99e4917bbe9f37f8e
Gerrit URL: [https://gerrit.gromacs.org/3491](https://gerrit.gromacs.org/3491)

#5 - 05/23/2014 01:50 PM - Rossen Apostolov

I added only a warning because in most cases it will work fine despite the memory requirements.

#6 - 06/12/2014 01:59 AM - Erik Lindahl

- Target version changed from 5.0 to future

#7 - 12/31/2017 12:10 AM - Erik Lindahl

- Status changed from New to Closed

05/02/2020
No point in keeping this open. As the C++ underpinnings are maturing it's time to reimplement many of these tools, and then we should be strict about not allowing any memory leaks. However, we won't touch the old tools.