

## Aufgabe 3)

- Die erste Möglichkeit einen oder mehrere Parameter an ein Programm zu übergeben, dass mit ddt gedebuggt wird ist direkt im Terminal

ddt ./programm-name *argument<sub>1</sub>* *argument<sub>2</sub>* ... *argument<sub>n</sub>*

einzugeben.

Die andere ist es nach dem man ddt im Terminal aufgerufen hat die Parameter mittels der GUI in das Arguments Textfeld einzutragen.

- Es gibt insgesamt 6 unterschiedliche Möglichkeiten das Programm fortzuführen.
  - Run/Continue:  
Startet das Programm bzw. setzt es nach einem Start und dem halten an einem Breakpoint fort. Ähnlich der gdb Funktion continue.
  - Step Into:  
Springt in eine Funktion um diese genauer Debuggen zu können. Ähnlich der gdb Funktion step.
  - Step Over:  
Überspringt eine Funktion, bei der der Debugger normalerweise hineingestept wäre. Ähnlich zu der gdb Funktion next.
  - Step Out:  
Springt aus einer Funktion in die hinein gestept wurde heraus, ähnlich zu der gdb Funktion finish
  - Run to Line:  
Führt das Programm solange aus, bis eine bestimmte Zeile erreicht wurde. Ähnlich der gdb Funktion break "*linenumber*" und dann continue
- Bei anschauen der Variable rank im Programm existieren mehrere Möglichkeiten an diese zu gelangen, hierbei gibt es:
  - Das Kontextmenü welches erscheint, wenn man die Maus über den Namen der Variable hält. Hierbei erscheint sowohl der Typ als auch der Name, so wie der aktuelle Wert der Variable.
  - Alternativ können die gleichen Informationen über das rechte Menu Locals erhalten werden (aber nur, wenn sich der Breakpoint auch innerhalb des gleichen Scopes ist)
  - Eine weitere Alternative ist Current Line, bei der aber die Variable innerhalb der ausgewählten Lines sein muss.

Wichtig hierbei ist allerdings, dass nur die Variable des aktuell ausgewählten Prozesses betrachtet werden kann, um die anderen Werte der Prozesse zu betrachten, muss der Prozess gewechselt werden. Dies kann über einen Rechtsklick auf die Variable gelöst werden. Hierbei erscheint ein Kontextmenü wo im oberen Eintrag der Name der Variable erscheint und dort Submenüs aufgerufen werden können, hierbei existiert ein Eintrag Compare Across Processes bzw. Compare Across Threads über den die Variable über alle Prozesse und / oder Threads verglichen werden kann.

Eine Andere Möglichkeit wäre den Prozess zu wechseln über die im Hauptmenü eingetragenen zahlen 0 bis Prozesszahl und dann die Variable erneut zu betrachten.

- Im Evaluate Fenster können Ausdrücke eingetragen werden wie zum Beispiel  $rank == 1$ , dann wird dieser Ausgewertet und eine Statistik zeigt wie viele Prozesse diese Bedingung erfüllen, hierbei sind alle mathematischen Operatoren möglich. Des Weiteren können die Adressen von Funktionen angezeigt werden, so wie werte über alle Prozesse und / oder Threads verglichen werden. Ein Weiteres Feature ist, dass der Wert der betrachteten Variable verändert werden kann.
- Das evaluate Fenster evaluiert für jeden Prozess die Ausdrücke neu. Beispiel:  
Wenn zwei Prozesse gestartet werden und diese den rank 0 bzw 1 haben und man nun im Evaluate Fenster folgendes evaluieren möchte,  $rank < 1$  dann ist dies im Prozess 0 wahr und im Prozess 1 falsch. Es wird also jedes mal die Ausdrücke neu ausgewertet.
- Die erste Möglichkeit besteht darin, dass man sich das Array wieder über das Kontextmenü anschaut, welches beim Rechtsklick erscheint. Hierbei wird das Array über ein Extra Evaluate Fenster ausgewertet, in dem es auch möglich ist, dass mehrdimensionale Arrays angezeigt werden können. Alternativ kann man sich das Array wieder über die schon beschriebenen Funktionen Current Line bzw. Locals anschauen und bekommt diese dann in einer Liste präsentiert.