

## Übungen zur Computerorientierten Physik

### 8 Ereignis getriebene Simulationen

1. Laden Sie sich das Programm `chain_heap_fragment.c` vom StudIP.
2. Lernen Sie das Programm gut kennen. Studieren Sie dazu
  - Die Datenstrukturen `particle_t`, `event_t`, `heap_elem_t` und `global_t` am Anfang.
  - Den Aufbau der Hauptroutine `main()`
  - Die Aufrufstruktur der Funktionen (Parameter) ihre Funktion (Kommentar am Funktionskopf)
  - Die Implementierung der Funktion `treat_event()`
  - Die Implementierung der Funktion `heap_remove()`, die ein Element aus dem Heap entfernt. Beachten Sie insbesondere, wie bei jeder Positionsänderung eines Elements im Heap auf die Referenz `event[...].pos` von dem Ereignis in den Heap mit aktualisiert wird.

Hinweis: Das Programm kann ohne weitere Source Files (nur Flag `-lm` strikt nötig) compiliert werden und wird aufgerufen mit

```
chain_heap_fragment <#particles> <t_max>
```

3. Implimentieren Sie die Funktion

```
/****** heap_insert() *****/
/** Insert event 'event[ev]' into heap 'glob->heap' **/
/** (and stores position in heap in event **/
/** PARAMETERS: (*)= return-paramter **/
/**      glob: global data **/
/**      event: array of events **/
/**      ev: id of event **/
/** RETURNS: **/
/**      nothing **/
/******/
void heap_insert(global_t *glob, event_t *event, int ev)
```

indem Sie den vorgegebenen Rahmen komplettieren. Tipps: In `heap_remove()` können Sie sehen wie das kompliziertere Entfernen aus dem Heap funktioniert und das als Vorlage benutzen. Beachten Sie, dass das Feld `pos` bei allen betroffenen Ereignissen angepasst wird. (6 Punkte)

4. Testen Sie die Funktion, indem Sie mit dem Debugger `gdb` verfolgen, wie einzelne Elemente eingesetzt werden.

Setzen Sie dazu einen Breakpoint auf die erste Zeile der Funktion, und führen Sie die Funktion schrittweise aus, wenn der Debugger am Breakpoint anhält. Lassen Sie sich immer die Variablenwerte ausgeben um zu überprüfen, ob das Ergebnis korrekt ist.

(1 Punkt)

5. Lassen Sie Simulationen laufen, und messen Sie Dichtekurven für 1000 Teilchen und Endzeit  $t = 10^5$  für

- a) Teilchen, die alternierend die Masse 1 und 2.4 haben (wie in Vorlesung)
- b) Teilchen, die alternierend die Masse 1 und 100 haben
- c) Teilchen, die alternierend die Masse 1 und 1.1 haben
- d) Teilchen, die alle die gleiche Masse haben.
- e) Teilchen, deren Massen zufällig in  $[1, 10]$  verteilt sind.

(Welche Funktion müssen sie für b), c), d), e) modifizieren?)

und stellen Sie jeweils die Kurven mit `gnuplot` dar.

(3 Punkte)