**Prozesse, Threads**

Programm ist ein Rezept.

Prozess ist ein Programm in Ausführung.

Prozess hat einen bestimmten Bereich im Hauptspeicher.

Text, Data statisch. Heap und Stack dynamisch, können größer/kleiner werden aber keinen max.-wert überschreiten.

Stack enthält Methoden. (Call Stack)

Stack hat nur 2 Zugriffsmethode (Push, Pop).

Push nach oben hin anfüllen. Pop nimmt oberes Element heraus. (LIFO, Last in First out)

Auf ein bestimmtes Element zuzugreifen ist nicht möglich.

Rekursion StackOverflowError, wenn Rekursion nicht abgebrochen wird und der Stack im Hauptspeicher zu groß wird.

Kein trycatch, weil Error und keine Exception, Programm steht.

**Stack**: Methoden (Methoden Aufruf, Variablen in den Methoden) (primitive Datentypen)

**Data**: globale und static Variablen (bei Objecte wieder eine Referenz in den Heap).

**Heap**: (Objekte (new)) (z.B.: anlegen eines Arrays, im Stack jedoch dann die Referenz auf das Objekt/Array)

**Text/Code**: Assembler Befehle

**Prozesszustände:**

Zustand 1: new (nimmt viel Zeit in anspruch) Phase wo Betriebssystem Platz schafft. Ein Prozess wurde erzeugt. Und den ganzen Rest für den Prozess initialisiert.

Zustand 2: ready Programm läuft noch nicht.

Zustand 3: running Programm läuft

(durch scheduling wechsel zwischen ready und running)

Zustand 4: terminated, der Prozess wurde abgeschlossen und beendet (Alle Assembler Befehler abgeschlossen).

Zustand 5: waiting, der Prozess wartet (z.B.: Auf eingabe des Benutzers oder auf ein bestimmtes Event)