

Aufgabe 1

Erläutern Sie den Unterschied zwischen einem Prozess, einem Thread und einem Programm.

Ein Programm besteht aus Code und dient dazu, eine bestimmte Funktionalität zu erfüllen. Wenn man dieses Programm startet, erzeugt das Betriebssystem einen Prozess für das Programm, indem es einen Hauptthread initialisiert, der den Code ausführt. Dieser Hauptthread kann weitere Threads erzeugen, um Code parallel auszuführen.

Aufgabe 2

Berechnen Sie den Speedup eines Programms mit Hilfe des Amdahlschen Gesetzes, wenn der parallelisierbare Teil des Programms bei 25, 50 und 75 Prozent liegt. Berechnen Sie für 2^x Kerne mit $x = 0$ bis $x \leq 3$.

25% parallelisierbarer Teil:

$$2^0 = 2 \text{ Kerne} \quad 0,75 + \frac{1-0,75}{2} = 0,875 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,875 = 1,14$$

$$2^1 = 4 \text{ Kerne} \quad 0,75 + \frac{1-0,75}{4} = 0,813 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,813 = 1,23$$

$$2^2 = 8 \text{ Kerne} \quad 0,75 + \frac{1-0,75}{8} = 0,781 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,781 = 1,28$$

$$2^3 = 16 \text{ Kerne} \quad 0,75 + \frac{1-0,75}{16} = 0,766 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,766 = 1,31$$

50% parallelisierbarer Teil:

$$2^0 = 2 \text{ Kerne} \quad 0,5 + \frac{1-0,5}{2} = 0,75 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,75 = \frac{4}{3}$$

$$2^1 = 4 \text{ Kerne} \quad 0,5 + \frac{1-0,5}{4} = 0,625 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,625 = 1,6$$

$$2^2 = 8 \text{ Kerne} \quad 0,5 + \frac{1-0,5}{8} = 0,563 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,563 = 1,78$$

$$2^3 = 16 \text{ Kerne} \quad 0,5 + \frac{1-0,5}{16} = 0,531 \text{ s} \quad \text{Ausführungszeit, also Speedup: } 1/0,531 = 1,88$$

75% parallelisierbarer Teil:

$2^0 = 2$ Kerne $0,25 + \frac{1-0,25}{2} = 0,625 s$ Ausführungszeit, also Speedup: $1/0,625 = 1,6$

$2^1 = 4$ Kerne $0,25 + \frac{1-0,25}{4} = 0,438 s$ Ausführungszeit, also Speedup: $1/0,438 = 2,28$

$2^2 = 8$ Kerne $0,25 + \frac{1-0,25}{8} = 0,344 s$ Ausführungszeit, also Speedup: $1/0,344 = 2,91$

$2^2 = 16$ Kerne $0,25 + \frac{1-0,25}{16} = 0,297 s$ Ausführungszeit, also Speedup: $1/0,297 = 3,37$

Aufgabe 3)

Siehe java-Programm Counterparallel.java

Ich habe das Programm mit `.parallel()` im Stream parallelisiert. Mit `ForkJoinPool` kann ich kontrollieren wie viele Threads mit `parallel()` erzeugt werden.

Dann hab ich einen Ordner auf meinem PC nach `.java` dateien mit dem Programm durchsucht. Mit den Programm-Argumenten: `C:\Users\Maxim\Desktop\MaximsZeug ".*\java"`

ich erhielt folgende Ausgaben:

mit Threads = 1: Total Lines: 1250
 Time Taken: 1.009 seconds

mit Threads = 2: Total Lines: 1250
 Time Taken: 0.566 seconds

mit Threads = 4: Total Lines: 1250
 Time Taken: 0.433 seconds

mit Threads = 8: Total Lines: 1250
 Time Taken: 0.432 seconds

mit Threads = 16: Total Lines: 1250
 Time Taken: 0.433 seconds

Also ab 4 Threads kein bedeutender Unterschied mehr. Man beachte aber auch das 1250 Lines eine sehr kleine Menge ist und bei mehr Dateien der Unterschied wahrscheinlich größer wäre.