Opdrachten week 6 – Schedulability Analyses

6.1 Een programma bestaat uit 4 taken T_1 t/m T_4. Deze taken gebruiken geen gedeelde resources. In de tabel 1 staat i voor het nummer van de taak, T_i voor de periodetijd van taak i en C_i voor de maximale executietijd van taak i. Gegeven is dat de deadline van elke taak gelijk is aan zijn periodetijd.

Tabel 1: De gegevens van de taken

i	T_i	C_i
1	100	50
2	280	45
3	200	20
4	300	40

Alle gegeven tijden zijn in ms.

A Bepaal de schedulability van deze taken met behulp van de "Utilization based schedulability test". Geef de benodigde berekening en trek daaruit je conclusie!

De test is:

$$\sum_{i}^{N} \frac{C_i}{T_i} \leq N\left(2^{\frac{1}{N}} - 1\right)$$

In dit geval hebben we 4 taken dus $N = 4$. We kunnen de sommatie nu uitschrijven:

$$\frac{C_1}{T_1} + \frac{C_2}{T_2} + \frac{C_3}{T_3} + \frac{C_4}{T_4} \leq 4\left(2^{\frac{1}{4}} - 1\right)$$

Invullen van alle overige waarden geeft:

$$\frac{50}{100} + \frac{45}{280} + \frac{20}{200} + \frac{40}{300} \leq 4\left(2^{\frac{1}{4}} - 1\right)$$

Als we de waarde links en rechts uitrekenen vinden we:

$$0,894 \leq 0,757$$

De test is dus niet geslaagd. We weten dus niet zeker of alle deadline behaald zullen worden. Maar omdat de processor maar voor 89,4 % wordt benut (utilisation) zou dit wel kunnen. Om te bepalen of alle deadlines worden behaald moeten de responsetijden van alle taken worden bepaald. Zie opdracht C.

B Bepaal de prioriteiten P_i van de verschillende taken als gebruik gemaakt wordt van FPS-RMPA (Fixed-priority Pre-emptive Scheduling – Rate Monotonic Priority Assignment). Het systeem kent 4 verschillende prioriteiten (1 t/m 4) waarbij 4 de hoogste prioriteit is.

Rate Monotonic Priority Assignment: $T_i < T_j \implies P_i > P_j$. Zie tabel 2.

Tabel 2: De prioriteiten van de taken

i	T_i	C_i	P_i
1	100	50	4
2	280	45	2
3	200	20	3
4	300	40	1

C Bereken voor alle taken of de deadline wordt gehaald en geef, indien de deadline wordt gehaald, de response tijd R_i.

$$R_i = C_i + \sum_{j \in hp(i)} \left\lceil \frac{R_i}{T_j} \right\rceil C_j$$

Het probleem bij deze formule is dat R_i links en rechts van het =-teken staat. Dit is namelijk niet te vereenvoudigen omdat de ceiling functie geen inverse functie heeft. We lossen dit op door de formule herhaaldelijk in te vullen:

$$\omega_i^{n+1} = C_i + \sum_{j \in hp(i)} \left\lceil \frac{\omega_i^n}{T_j} \right\rceil C_j$$

Hierin is ω_i^n de n-de iteratie van ω_i. We stoppen als $\omega_i^{n+1} = \omega_i^n$, dan geldt $R_i = \omega_i$, of als $\omega_i^{n+1} > D_i$, omdat de deadline dan wordt overschreden. Omdat in dit geval gegeven is $D_i = T_i$, stoppen we als $\omega_i^{n+1} > T_i$.

We zullen hier, als voorbeeld, R_2 berekenen: $i = 2$ dus:

$$\omega_2^{n+1} = C_2 + \sum_{j \in hp(2)} \left\lceil \frac{\omega_2^n}{T_j} \right\rceil C_j$$

$hp(2)$ is de verzameling taken met een hogere prioriteit dan taak 2. In dit geval zijn dit taak 3 en 1, dus $hp(2) = \{3, 1\}$. Dus:

$$\omega_2^{n+1} = C_2 + \sum_{j \in \{3,1\}} \left\lceil \frac{\omega_2^n}{T_j} \right\rceil C_j$$

We kunnen nu de sommatie uitschrijven:

$$\omega_2^{n+1} = C_2 + \left\lceil \frac{\omega_2^n}{T_3} \right\rceil C_3 + \left\lceil \frac{\omega_2^n}{T_1} \right\rceil C_1 \tag{1}$$

We weten zeker dat $R_2 \geq 0$, want een taak kan niet reageren voordat hij gestart is. Dus starten we de iteratieve berekening, voor $n = 0$, met $\omega_2^0 = 0$. Alle waarden kunnen nu worden ingevuld:

$$\omega_2^1 = 45 + \left\lceil \frac{0}{200} \right\rceil 20 + \left\lceil \frac{0}{100} \right\rceil 50 = 45$$

Omdat $\omega_2^1 \neq \omega_2^0$ ($45 \neq 0$) en $\omega_2^1 \leq T_2$ ($45 \leq 280$) vullen we vergelijking (1) nogmaals in, maar nu voor $n = 1$:

$$\omega_2^2 = 45 + \left\lceil \frac{45}{200} \right\rceil 20 + \left\lceil \frac{45}{100} \right\rceil 50 = 115$$

Omdat $\omega_2^2 \neq \omega_2^1$ ($115 \neq 45$) en $\omega_2^2 \leq T_2$ ($115 \leq 280$) vullen we vergelijking (1) nogmaals in, maar nu voor $n = 2$:

$$\omega_2^3 = 45 + \left\lceil \frac{115}{200} \right\rceil 20 + \left\lceil \frac{115}{100} \right\rceil 50 = 165$$

Omdat $\omega_2^3 \neq \omega_2^2$ ($165 \neq 115$) en $\omega_2^3 \leq T_2$ ($165 \leq 280$) vullen we vergelijking (1) nogmaals in, maar nu voor $n = 3$:

$$\omega_2^4 = 45 + \left\lceil \frac{165}{200} \right\rceil 20 + \left\lceil \frac{165}{100} \right\rceil 50 = 165$$

Nu geldt $\omega_2^4 = \omega_2^3$ ($165 = 165$) dus $R_2 = 165$. Omdat $R_2 \leq T_2$ ($165 \leq 280$) kunnen we concluderen dat taak 2 zijn deadline (ruim) haalt.

Op vergelijkbare wijze kunnen we de responsetijden van de overige taken bepalen, zie tabel 3.

Tabel 3: De responsetijden van de taken

i	T_i	C_i	P_i	R_i	**Deadline behaald?**
1	100	50	4	50	ja
2	280	45	2	165	ja
3	200	20	3	70	ja
4	300	40	1	275	ja