

23-10-2018_PS

I. Aufgabe 1

II. Aufgabe 2

III. Aufgabe 3

Aufgabe 1

Wir rechnen aus, wieviele Tupel in einem Block sind:

Blockgrosse $B = 8192$, Tupelgrosse $t = 120$. $\frac{B}{t} = 68$ Tupel pro Block (abgerundet).

Wir rechnen weiter aus, wieviele Bloেকে die Relationen brauchen: $|R| = |S| = 1.000.000$ und $\frac{|R|}{68} = 14.706$

Fuer $\sigma_{A \neq 7.000.000}(R)$ und $\sigma_{A \neq 7.000.000}(S)$: Wir lesen einfach die ganze Relation undn schmeissen unpassende Tupel raus. Bei 0,02s fuer einen Lesezugriff erhalten wir $14.706 * 0.02 = 294s$. Es hilft nichts, dass S sortiert ist.

Fuer $\sigma_{A < 7.000.007}(R)$ muessen wir auch alle Bloেকে lesen (und kommen wieder auf 294), das aendert sich aber bei S : Da S sortiert ist, koennen wir mit Binarysearch das Tupel 7.000.007 finden und dann alle Werte "darunter" ausgeben. Binarysearch ist logarithmisch und wir vernachlaessigen die Laufzeit davon: $\frac{7.000.007}{5.000.000} \approx 40\%$ und $14706 * 0.4 = 5882,4$ und $14706 * 0.4 = 5882,4$ und $14706 * 0.4 = 5882,4$. Wir koennen nur ganze Bloেকে lesen und runden daher auf: $\lfloor 5882,4 \rfloor = 5883$.

Aufgabe 2

Unsere Slotted Page faengt an mit a , f als Pointer zum Ende des freien Platzes, q_n und p_n halten Details zu jedem Tupel fest. Die Datensatzgrosse ist d_n und wir rechnen zunaechst die Loesung fuer $|d_n| = 2^5$ aus.

Es gilt $|a| = |f| = |q_n| = |p_n| = 13$. Wir stellen eine Formel auf:

$$|a| + |f| + n(|q_n| + |p_n| + |d_n|) \leq 2^{13} * 8$$

Wir setzen ein und vereinfachen:

$$13 + 13 + n(13 + 13 + 2^5 * 8) \leq 2^{13} * 8$$

$$13 + 13 + n(13 + 13) \leq 2^{13} * 8$$

$$n \leq \frac{2^{13} * 8 - 2 * 13}{2 * 13 + 2^5 * 8} = 232$$

Man kann also **232** Datensaeetze speichern.

Aufgabe 3

Wir haben $\frac{8192}{2} = 4096$ Datensätze mit Adressen 0 bis 4095. Wir finden die Größe unserer Datensätze:

$|(1, 'Alpha')| = 4 + 5 = 9$ und $|(2, 'Pi')| = 4 + 2 = 6$ und $|(3, 'Epsilon')| = 4 + 7 = 11$. Ungerade Größen werden aufgerundet, da unsere Adressen durch 2 teilbar sein müssen: Wir erhalten **10, 6 und 12 Bytes** als Tupelgrößen, die Datensätze nehmen also 5, 3 und 6 Wörter ein.

Unsere Slotted Page besteht dann aus $a, f, q_{1...3}, p_{1...3}, d_{1...3}$. Die Adressen, an denen wir die Datensätze ablegen, finden wir, indem wir sukzessive für jeden Datensatz von unserem Free-Space-Pointer f die Anzahl eingenommener Wörter des jeweiligen Datensatzes abziehen. f und a ändern wir nach jeder Insertion entsprechend.

Am Ende sieht unsere Slotted Page so aus:

a	f	q_1	p_1	q_2	p_2	q_3	p_3	...freier Speicher...	Datensatz 3	Datensatz 2	Datensatz 1
3	4018	9	4091	6	4038	12	4082	...	(3, 'Epsilon')	(2, 'Pi')	(1, 'Alpha')