Homework 6

Alexander Björkman
(albjorkm@kth.se, Plusplus)

1 Induktion

Vi vill bevisa följande uttryck med hjälp av induktionens principer:

$$\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$$

Vi börjar med att kolla ifall det stämmer för $n = 1$. För vänster led får vi då:

$$VL = 1^2 = 1$$

Medan för höger led får vi:

$$HL = \frac{1(1+1)(2 \cdot 1 + 1)}{6} = \frac{1(2)(3)}{6} = \frac{6}{6} = 1$$

Vi noterar att $VL = HL$ och därmed stämmer det för basfallet. Vi antar att uttrycket är sannt för $n = k$. Vi vet då att:

$$\sum_{i=1}^{k} i^2 = \frac{k(k+1)(2k+1)}{6}$$

Vi vill nu bebisa att likheten även gäller för $n = k + 1$. Vi vill att VL = HL för $n = k + 1$. Enligt ursprungliga likheten är höger led:

$$HL = \frac{(k+1)(k+1+1)(2(k+1)+1)}{6} = \frac{(k+1)(k+2)(2k+3)}{6}$$

$$HL = \frac{(k+1)(2k^2+3k+4k+6)}{6} = \frac{(k+1)(2k^2+7k+6)}{6}$$

Vi börjar med VL nu.

$$VL = \sum_{i=1}^{k+1} i^2 = \sum_{i=1}^{k} i^2 + (k+1)^2$$

Vi använder faktumet att likheten antas vara sann för $n = k$.

$$VL = \sum_{i=1}^{k} i^2 + (k+1)^2 = \frac{k(k+1)(2k+1)}{6} + (k+1)^2$$

Vi faktorerar ut $(k+1)$:

$$VL = (k+1)(\frac{k(2k+1)}{6} + (k+1))$$
$$VL = (k+1)(\frac{(2k^2+k)+(6k+6)}{6})$$
$$VL = (k+1)(\frac{(2k^2+7k+6)}{6})$$
$$VL = \frac{(k+1)(2k^2+7k+6)}{6}$$
$$VL = \frac{(k+1)(2k^2+7k+6)}{6}$$

Svar: Vi inser att $VL = HL$ och därmed enligt induktionenes principer är det bevisat.

2 Iterativ korrekhet

2.1 Korrekthet

Funktionen är ej korrekt. Ifall $n = -1$ hade en korrekt implementation of exponent retunerna $1/x$. Denna implementation retunerar istället 1.0 för vilket värde som helst av x. Vi kan bevisa detta genom att kompiler koden och köra den. Alternativt granskar vi koden och inser att algoritmen initialiseras med variabeln res satt till 1.0. Då $n = -1$ kopper loopen sedan omedelbart att termineras utan att någonsin påverkat funktionens state. Res är sedan retunerat. Alltså retunerar funktionen alltid 1.0 för $n = -1$. Det går att argumentera för att uppgiften antyder på att $n >= 0$. Men isåfall hade proceduren haft följande signatur istället:

```
double expIterative(double x, unsigned int n);
```

Svar: Vi har visat att funktionen är inkorrekt genom att ange ett motexempel.

2.2 Tidskomplexitet

expIterative är en mycket enkel funktion som endast består av initialisering, return och en loop invariant. Loop invariant sker n gånger (d.v.s argumentet n till funktionen) och utför en konstant mängd operationer (multiplikation av doubles, multiplikation av bignum har ej O(1) tidskomplexitet). Därmed är tidskomplexiteten $O(n)$.

3 Rekursiv korrekthet

3.1 Korrekthet

Denna funktion är ej korrekt utav samma anledning som den iterativa lösningen. Vi bevisar det genom att ange det inkorrekta fallet av:

```
expRecursive(2, -1)
```

Enligt exponents definition förväntas $2^-1 = 0.5$. Men då expRecursive anropar expIterative för alla värden mindre eller lika med 4, uppstår liknande problem som beskrevs för den iterativ lösningen.

3.2 Tidskomplexitet

Mästarsatsen defineras som:

$$T(n) = aT(\frac{n}{b}) + f(n)$$

Där $a = 2$, $b = 2$ och $f(n) \in \Theta(1)$ ger oss $T(n) \in \Theta(n)$. Alltså har den samma tidskomplexitet som den interativa metoden. Anledningen till detta är att även ifall problemet halveras per steg, så får vi två stycken halvor varje steg. Detta jämnar ut sig till “en hel” igen. Problemet halveras dock inte riktigt. Utan den blir lite mindre och lite större för än halva pga avrudning. Men detta jämnas ut utav att de aviker från hälften med lika mycket fast åt motsat riktning.