# DD2350 Algoritmer, datastrukturer och komplexitet, hösten 2019
# Lösningsförslag till övningsmästarprov 2: komplexitet

**Lika vikt**

Först visar vi att likaviktsproblemet ligger i NP. För varje ja-instans ska en lösning kunna verifieras i polynomisk tid. Låt lösningen vara en partitionering av dom $n$ prylarna i två disjunkta mängder. Låt partitioneringen representeras av en boolesk array $A[1..n]$ där $A[i]$ är sann om pryl nummer $i$ är i första mängden och falsk om den är i andra mängden. Verifieringsalgoritmen behöver bara summera vikterna i vardera mängden och kontrollera att summorna är lika.

```
EqualWeightVerify({v_i}_1^n, A[1..n]) =
        sum1 ← 0
        sum2 ← 0
        for i ← 1  to n do
            if A[i] then sum1 ← sum1 + v_i
            else sum2 ← sum2 + v_i
        return sum1 = sum2
```

Tidskomplexiteten för verifieringsalgoritmen är $O(n)$ med enhetskostnad och $O(nm)$ med bitkostnad (där $m$ är antalet bitar i indata), vilket är polynomiskt.

Vi visar nu att problemet är NP-svårt med en reduktion av problemet delmängdssumma. Låt $\sigma$ vara summan av alla tal i $P$. Idén är att vi låter vikterna motsvara talen i indata $P$. Om den sökta summan $K = \sigma/2$ så kommer en uppdelning av prylarna i två delar att ge delar med just vikten $K$. Annars lägger vi till en extra pryl med vikt $\Delta$ som är avpassad så att en uppdelning av prylarna i två delar med lika vikt motsvarar att det finns en delmängd av $P$ med den sökta summan $K$.

Hur stor ska vikten $\Delta$ vara? Om vi kan partitionera dom $n+1$ prylarna i två lika delar så väger varje del $(\sigma+\Delta)/2$. Om $K > \sigma/2$ så vill vi att ena delen ska ha vikt $K$. Ekvationen $K = (\sigma+\Delta)/2$ har lösningen $\Delta = 2K - \sigma$. Om $K < \sigma/2$ så vill vi att ena delen ska ha vikt $K + \Delta$. Ekvationen $K + \Delta = (\sigma + \Delta)/2$ har lösningen $\Delta = \sigma - 2K$.

Vi låter alltså $\Delta$ vara $|2K - \sigma|$.

I pseudokod blir reduktionen så här:

```
Delmängdssumma({p_i}_1^n, K) =
    σ ← 0
    for i ← 1  to n do
        v_i ← p_i
        σ ← σ + p_i
    if K ≠ σ/2 then v_{n+1} ← |2K − σ|
    return LikaVikt(v[])
```

För att bevisa att detta är en Karpreduktion ska vi visa att varje ja-instans för delmängdssumma transformeras till en ja-instans för likaviktsproblemet och att varje nej-instans för delmängdssumma transformeras till en nej-instans för likaviktsproblemet. Den andra delen är ekvivalent med att visa att för varje ja-instans för likaviktsproblemet som kan skapas av transformationen så är motsvarande instans av delmängdssumma en ja-instans.

I en uppgift på E-nivå krävs inte att dessa två bevis genomförs i detalj, men så här kan dom se ut:

1. Om det finns en delmängd $S \subseteq P$ med summa $K$ så ska vi visa att det finns en partitionering av $v[]$ i två delar med lika vikt. Låt $\sigma$ vara summan av talen i $P$.

   Fall 1: Om $K = \sigma/2$ så kommer både $S$ och $P - S$ att ha samma summa, varför motsvarande partitionering ger lika vikt.

Fall 2: Om $K > \sigma/2$ så låter vi $S$ vara ena partitionen (med vikt $K$), och den andra partitionen får då vikt $\Delta + \sigma - K = 2K - \sigma + \sigma - K = K$, alltså lika vikt.

Fall 3: Om $K < \sigma/2$ så låter vi både $\Delta$ och elementen i $S$ vara i ena partitionen som då får vikt $\Delta + K = \sigma - 2K + K = \sigma - K$, vilket är precis vad andra partitionen har som vikt.

I alla tre fallen finns det alltså en partitionering med lika vikt.

2. Anta nu istället att det finns en partitionering med lika vikt och visa att det måste finnas en delmängd med summa $K$. Vi har samma tre fall som ovan.

   Fall 1: Om $K = \sigma/2$ så kommer partitionernas vikt att vara precis $\sigma/2 = K$ och alla element kommer från $P$, så det finns en delmängd (till och med två delmängder) med summa $K$.

   Fall 2: Om $K > \sigma/2$ så kommer partitionernas vikt att vara $(\Delta+\sigma)/2 = (2K-\sigma+\sigma)/2 = K$. I den ena partitionen finns bara element från $P$, så det finns alltså en delmängd med summa $K$.

   Fall 3: Om $K < \sigma/2$ så är $\Delta = \sigma - 2K$, dvs $\sigma = \Delta + 2K$. Partitionernas vikt kommer att vara $(\Delta + \sigma)/2 = (\Delta + \Delta + 2K)/2 = \Delta + K$. I den partition som $\Delta$ tillhör kommer alltså övriga element (som vara kommer från $P$) att ha vikten $K$, som önskat.

   I alla tre fallen finns det alltså en delmängd med summa $K$.

Det är också nödvändigt att visa att Karpreduktionen tar polynomisk tid. Varje varv i slingan i reduktionen tar linjär tid i antalet bitar i dom ingående vikterna. Om antalet bitar i indata är $m$ så tar reduktionen tid $O(nm)$ med bitkostnad, vilket är polynomiskt. Alltså är likaviktsproblemet NP-fullständigt.

Problemet lika vikt är mer känt som *mängdpartitionering* (engelska *Partition*).