Talnafræði

Veldishafning

Bergur Snorrason

March 11, 2024

- Í talnafræðidæmum er algengt að skila eigi leif svars.

- Í talnafræðidæmum er algengt að skila eigi leif svars.
- Þetta er gert því leif er takmörkuð, en í talningarfræði er algengt að fá mjög stórar tölur.

- Í talnafræðidæmum er algengt að skila eigi leif svars.
- Þetta er gert því leif er takmörkuð, en í talningarfræði er algengt að fá mjög stórar tölur.
- Tökum dæmi.

Veldishafning

- Gefnar eru þrjár jákvæðar heiltölur x, n og m.

Veldishafning

- Gefnar eru þrjár jákvæðar heiltölur x, n og m.
- Finnið $x^n \mod m$.

- Þetta er lítið mál að gera í $\mathcal{O}(n)$ tíma.

- Þetta er lítið mál að gera í $\mathcal{O}(n)$ tíma.

```
22 ll modpow_linear(ll x, ll n, ll m)
23 {
24     ll r = 1;
25     while (n--) r = (r*x)%m;
26     return r;
27 }
```

- Þetta er lítið mál að gera í $\mathcal{O}(n)$ tíma.

```
22 ll modpow_linear(ll x, ll n, ll m)
23 {
24     ll r = 1;
25     while (n--) r = (r*x)%m;
26     return r;
27 }
```

- Við getum þó leyst þetta hraðar.

- Þetta er lítið mál að gera í $\mathcal{O}(n)$ tíma.

```
22 ll modpow_linear(ll x, ll n, ll m)
23 {
24     ll r = 1;
25     while (n--) r = (r*x)%m;
26     return r;
27 }
```

- Við getum þó leyst þetta hraðar.
- Sú lausn byggir á að deila og drottna.

- Þetta er lítið mál að gera í $\mathcal{O}(n)$ tíma.

```
22 ll modpow_linear(ll x, ll n, ll m)
23 {
24     ll r = 1;
25     while (n--) r = (r*x)%m;
26     return r;
27 }
```

- Við getum þó leyst þetta hraðar.
- Sú lausn byggir á að deila og drottna.
- Takið eftir að $x^{2n} = x^n \cdot x^n$ og $x^{2n+1} = x^n \cdot x^n \cdot x$.

- Þetta er lítið mál að gera í $\mathcal{O}(n)$ tíma.

```
22 ll modpow_linear(ll x, ll n, ll m)
23 {
24     ll r = 1;
25     while (n--) r = (r*x)%m;
26     return r;
27 }
```

- Við getum þó leyst þetta hraðar.
- Sú lausn byggir á að deila og drottna.
- Takið eftir að $x^{2n} = x^n \cdot x^n$ og $x^{2n+1} = x^n \cdot x^n \cdot x$.
- Því getum við í hverju skrefi helmingað veldisvísinn.

- Við getum útfært þetta endurkvæmt.

- Við getum útfært þetta endurkvæmt.

```
 7 ll modpow_rec(ll x, ll n, ll m)
 8 {
 9     if (n == 0) return 1;
10     ll r = modpow_rec(x, n/2, m);
11     r = (r*r)%m;
12     return n%2 == 0 ? r : (r*x)%m;
13 }
```

- Við getum útfært þetta endurkvæmt.

```
 7 ll modpow_rec(ll x, ll n, ll m)
 8 {
 9     if (n == 0) return 1;
10     ll r = modpow_rec(x, n/2, m);
11     r = (r*r)%m;
12     return n%2 == 0 ? r : (r*x)%m;
13 }
```

- Við getum líka gert þetta með einfaldri `for`-lykkju.

- Við getum útfært þetta endurkvæmt.

```
7  ll modpow_rec(ll x, ll n, ll m)
8  {
9      if (n == 0) return 1;
10     ll r = modpow_rec(x, n/2, m);
11     r = (r*r)%m;
12     return n%2 == 0 ? r : (r*x)%m;
13 }
```

- Við getum líka gert þetta með einfaldri `for`-lykkju.

```
15 ll modpow(ll x, ll n, ll m)
16 {
17     ll r;
18     for (r = 1; n > 0; n = n/2, x = (x*x)%m) if (n&1) r = (r*x)%m;
19     return r;
20 }
```

- Eins og sagt var áðan þá helmingast veldisvísirinn í hverju skrefi.

- Eins og sagt var áðan þá helmingast veldisvísirinn í hverju skrefi.
- Svo tímaflækjan er $\mathcal{O}(\quad)$.

- Eins og sagt var áðan þá helmingast veldisvísirinn í hverju skrefi.
- Svo tímaflækjan er $\mathcal{O}(\log n)$.