Talnafræði

Veldishafning

Bergur Snorrason

10. mars 2021

- Í talningarfræðidæmum er algengt að skila eigi leif svars.

- Í talningarfræðidæmum 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 talningarfræðidæmum 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.

```
4  int main()
5  {
6      ll x, n, m, r = 1;
7      scanf("%lld%lld%lld", &x, &n, &m);
8      while (n-- != 0) r = (r*x)%m;
9      printf("%lld\n", r);
10     return 0;
11 }
```

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

```
4  int main()
5  {
6      ll x, n, m, r = 1;
7      scanf("%lld%lld%lld", &x, &n, &m);
8      while (n-- != 0) r = (r*x)%m;
9      printf("%lld\n", r);
10     return 0;
11 }
```

- Við getum þó leyst þetta hraðar.

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

```
4  int main()
5  {
6      ll x, n, m, r = 1;
7      scanf("%lld%lld%lld", &x, &n, &m);
8      while (n-- != 0) r = (r*x)%m;
9      printf("%lld\n", r);
10     return 0;
11 }
```

- 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.

```
4  int main()
5  {
6      ll x, n, m, r = 1;
7      scanf("%lld%lld%lld", &x, &n, &m);
8      while (n-- != 0) r = (r*x)%m;
9      printf("%lld\n", r);
10     return 0;
11 }
```

- 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.

```
4  int main()
5  {
6      ll x, n, m, r = 1;
7      scanf("%lld%lld%lld", &x, &n, &m);
8      while (n-- != 0) r = (r*x)%m;
9      printf("%lld\n", r);
10     return 0;
11 }
```

- 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.

```
4  ll modpow(ll x, ll n, ll m)
5  {
6      if (n == 0) return 1;
7      ll r = modpow(x, n/2, m);
8      r = (r*r)%m;
9      return n%2 == 0 ? r : (r*x)%m;
10 }
```

- Við getum útfært þetta endurkvæmt.

```
4  ll modpow(ll x, ll n, ll m)
5  {
6      if (n == 0) return 1;
7      ll r = modpow(x, n/2, m);
8      r = (r*r)%m;
9      return n%2 == 0 ? r : (r*x)%m;
10 }
```

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

- Við getum útfært þetta endurkvæmt.

```
4  ll modpow(ll x, ll n, ll m)
5  {
6      if (n == 0) return 1;
7      ll r = modpow(x, n/2, m);
8      r = (r*r)%m;
9      return n%2 == 0 ? r : (r*x)%m;
10 }
```

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

```
4  ll modpow(ll x, ll n, ll m)
5  {
6      ll r = 1;
7      while (n > 0)
8      {
9          if (n%2 == 1) r = (r*x)%m;
10         n = n/2;
11         x = (x*x)%m;
12     }
13     return r;
14 }
```

- 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 $\mathcal{O}(\log n)$.