

# Perfektní hešování

Pavel Bezstarosti, Karolína Burešová, Michael Pokorný

Supervizor: doc. Ing. Jaromír Kukal, Ph. D.

# Cíle práce

- Prozkoumat současné možnosti hešování
- Vyhledat minimální perfektní hešovací funkce pro konkrétní údaje
- Implementovat některé obvyklé algoritmy generování perfektních hešovacích funkcí

# Perfektní hešování

- Hešování = funkce nad množinou klíčů  $K$
- 1 funkční hodnota na max. 1 klíč → perfektní hešovací funkce (PHF)
- Minimální PHF:  $|K|=|H|$
- Urychluje přístup k neměnným datům
  - Nebo pomocné datové struktury k indexování dynamických dat

# Studované hešovací funkce

$$h(key) = key \bmod Q$$

$$h(key) = A \cdot key \bmod Q$$

$$h(key) = \left\lfloor \frac{A}{key} \right\rfloor \bmod Q$$

# Metodika generování HF

- Hrubá síla
- Heuristiky a optimalizace
- V zásadě nevylepšitelná složitost

# CHM

- Czech, Havas, Majewski
- Nedeterministický algoritmus pro vyhledání MPHF
- Order-preserving
- Založeno na náhodných grafech
- Mnohem lepší časové výsledky než hrubá síla

```

bool kolize=false; int* cisla={vstup}; int
pocet_cisel={vstup};

bool table[]; int keys[]; int kolize1,kolize2;
int A, Q = pocet_cisel;
hash(key) = floor(A/key)%pocet_cisel;

uloz(int key)
{
    int h=hash(key);
    if (table[h])
    {
        kolize1=key;
        kolize2=keys[h];
        kolize=true;
    }
    keys[h]=key;
    table[h]=true;
}

posun()
{
    int dA1 = floor((1 - (A/kolize1)%1) * kolize1);
    int dA2 = floor((1 - (A/kolize2)%1) * kolize2);
    A+=min(dA1,dA2);
    while (hash(kolize1)==hash(kolize2)) A++;
}

```

```

main()
{
    quicksort(cisla);
    kolize=true;

    Amin = ((Q-2)*cisla[0]*cisla[Q-1]) /
        (cisla[Q-1]-cisla[0]);
    Amax=cisla[0];
    Pro každý klíč kromě prvního:
        Amax=NSN(cisla[i],Amax);
    Amax*=Q;

    kolize=true;
    Pro všechna A mezi Amin a Amax, dokud kolize:
        if (kolize && A!=dolniA) posun();
        vyprázdni tabulku;

    Pro každý klíč:
        ulož(klíč);
        nedošlo-li ke kolizi, další klíč;
}

```

# Významné letopočty

- Reciproční,  $A = 38\ 685\ 101$ ,  $Q = 17$

Index	key
0	995
1	1989
2	1348
3	1415
4	1609
5	1968
6	1848
7	1918
8	929
9	1212
10	1939
11	1620
12	1781
13	1993
14	863
15	1914
16	1945



# Prvočísla do 50

- Reciproční,  $A = 804\ 885$ ,  $Q = 15$

Index	key
0	23
1	11
2	19
3	37
4	29
5	3
6	17
7	2
8	7
9	13
10	47
11	41
12	5
13	43
14	31

# Nešťastná prvočísla

- Prvočísla končící na 13
- Vybrán vzorek pod 5000
- Použito reciproční hešování
  - $A = 16\ 973\ 294$ ,  $Q = 17$

Index	key
0	2713
1	1213
2	2213
3	4513
4	13
5	3613
6	3313
7	4813
8	2113
9	3413
10	1013
11	113
12	613
13	4013
14	313
15	1913
16	1613

# Závěr

- Prozkoumali jsme některé možnosti hešování
- Děkujeme FJFI ČVUT za organizování Fyzikálního týdne na Jaderce a našemu supervizorovi doc. Ing. Jaromíru Kukalovi Ph. D. za náměty, informace a rady.

# Děkujeme za pozornost

Jsou nějaké dotazy?