Fakultas Informatika
School of Computing
Telkom University
Think Big! Go Forward!

Telkom University

ANALISIS ALGORITMA

Week 05: Pengantar Struktur Data

PROGRAM PASCA SARJANA INFORMATIKA
FAKULTAS TEKNIK INFORMATIKA
UNIVERSITAS TELKOM

2022/2023

Struktur Data: Hash

Pengalamatan Langsung (Direct Addressing)

- Nilai data digunakan sebagai indeks/alamat memori
  - Ingat bagaimana data digunakan pada metoda Counting Sort
  - Pemetaan 1 → 1 dari data menjadi alamat memori
- Insert: Set status lokasi yang akan diisi menjadi terisi! O(1)
  - Ini kalau duplikasi tidak diperbolehkan, sebagai counter jika diperbolehkan (re. counting sort)
  - Memori disediakan untuk semua KEMUNGKINAN DATA, bukan sebanyak data
  - Setiap sel memori berukuran secukupnya, sebagai tag (Occ/Free) atau counter
- Delete: Reset status menjadi free atau kurangi counter! O(1)
- Search: Periksa apakah lokasi berstatus terisi atau tidak tidak nol! O(1)
- FindMax: ???, Halah, periksa semua lokasi???
- Alokasi memori terkait rentang maksimum nilai m, maka O(m)
  - Harus ada pengukuran utilisasi space, $\alpha = n / m$ dimana n adalah jumlah data

Fungsi Hash

- Pemetaan data kesuatu alamat memori
  - Mengurangi kebutuhan total alokasi memori
  - Tetapi, pasti akan ada resiko tabrakan pemetaan
  - Karena pemetaan n → 1 dari data ke lokasi memori
  - Semua lokasi memori harus bisa dipetakan agar terpakai, tidak ada blackhole/unusable space
- Co. operasi modulo: $h(k) = k \bmod m$
  - Jika **max(k)=bm**, maka **b** digit atas kunci tidak terpakai dalam pemetaan
  - Gunakan nilai prima untuk m
- Co. operasi perkalian: $h(k) = \lfloor m(k/A - \lfloor k.A \rfloor) \rfloor$, $0 < A < 1$
  - Konstanta Knuth, A=0.618
  - Tidak terlalu sensitif terhadap nilai m yang digunakan

Struktur Hash

- Operasi Insert, Delete, Search bergantung pada probabilitas tabrakan
- Jika jumlah tabrakan $O(p_l)$, untuk p_l operasi insert, dimana
  - $p_l = O(n)$, n adalah jumlah data tersimpan
  - p = total operasi SID pada Hash, dan juga $p = O(n)$
  - Maka rerata setiap operasi SID membutuhkan waktu O(1)
- Kasus worst case?
  - Ketika fungsi hash (selalu/dipaksa) tabrakan, maka **Θ(n)**
  - dan menjadi O(m) dimana m adalah besar memori teralokasi untuk hash

Fungsi Hash dan ReHash

- Penghindaran tabrakan?
  - Bergantung pada data
  - Bergantung pada distribusi hasil pemetaan
- Resolution ketika tabrakan?
  - rehash
  - chaining
- Rehash/Probing: $h(k,i) = (h(k) + h_i(k)) \bmod m$
  - Dapat mengelompok, dimana sebagian area kosong dan sebagian lagi penuh data
- Universal Hash: Collection of hash functions $h_i(k)$, i=1..H
  - Diketahui probabilitas tabrakan per fungsi $h_i(k)$ adalah $|H|/m$
  - Rerata tabrakan menjadi 1/m jika fungsi dipilih secara acak

Colision, Chain, dan Perfect Hash

- Nilai yang tabrakan disimpan dalam suatu list
- Untuk memperoleh O(1) pada worst case → Perfect Hash
  - Dua tingkat fungsi hash
    - Satu primer, fungsi h(k) yang biasa
    - Fungsi hash pada space sekunder untuk menangani tabrakan
  - Space sekunder otmatis diperbesar **kuadratik terhadap jumlah tabrakan**

Hash vs. Struktur Lain

Primitives	U-Array	O-Array	Linked List	O Linked List	D Linked List	BST	Hash
Search	O(n)	O(n)/O(lg n)	O(n)	O(n)	O(n)	O(lg n)/O(n)	O(1)#
FindMax	O(n)	O(1)	O(n)	O(n)/O(1)	O(1)	O(lg n)/O(n)	?
Insert	O(1)	O(n)	O(1)	O(n)	O(n)/O(1)	O(lg n)/O(1)*	O(1)#
Delete	O(n)	O(n)	O(n)	O(n)	O(1)	O(lg n)/O(1)*	O(1)#

*) Diluar proses SearchParent atau FindMax

#) Perhitungan rata-rata, bukan worst-case

Message Digest

- Hash juga digunakan untuk message digests
  - Rangkaian bits yang hampi unik untuk suatu data
- Berguna untuk memeriksa integritas data tersebut
- Berguna untuk keamanan data
- Co.
  - MD5 (Message Digest v5) 128 bits, by Ron Rivest, 1992 (RFC1321)
  - SHA256 (Secure Hash Algorithm) 256 bits, rev 2011 (RFC6234)