

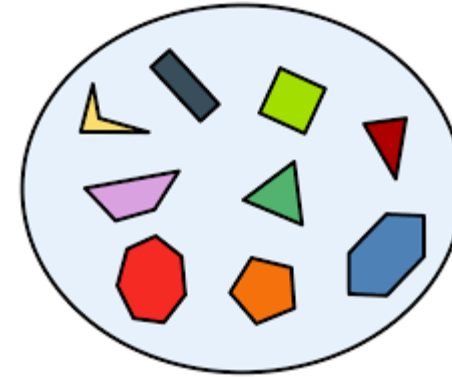
ADT Set

IF1210 – Algoritma dan Pemrograman 1
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Set – Definisi

Set adalah kumpulan objek yang:

- Memiliki tipe yang sama
- Setiap elemennya unik
- Elemen-elemennya tidak memiliki keterurutan: tidak ada istilah elemen 'next' dan 'previous'.



Set – Definisi Fungsional

Jika diberikan $S, S1, S2$ adalah **Set** dengan elemen $ElmtS$

<i>CreateSet</i> : $\rightarrow S$	{ Membuat sebuah set kosong }
<i>isEmpty</i> : $S \rightarrow \text{boolean}$	{ Tes terhadap S : true jika S kosong, false jika S tidak kosong }
<i>Length</i> : $S \rightarrow \text{integer}$	{ Mengirimkan banyaknya elemen S }
<i>add</i> : $ElmtS \times S \rightarrow S$	{ Menambahkankan $ElmtS$ ke S , jika $ElmtS$ belum terdapat di dalam S }
<i>remove</i> : $ElmtS \times S \rightarrow S$	{ Menghapus $ElmtS$ dari S }
<i>isIn</i> : $ElmtS \times S \rightarrow \text{boolean}$	{ Mengembalikan true jika $ElmtS$ ada di dalam S }
<i>isEqual</i> : $S1 \times S2 \rightarrow \text{boolean}$	{ Mengembalikan true jika $S1$ dan $S2$ memiliki elemen yang sama }
<i>union</i> : $S1 \times S2 \rightarrow S$	{ Menghasilkan gabungan $S1$ dan $S2$ }
<i>intersection</i> : $S1 \times S2 \rightarrow S$	{ Menghasilkan irisan $S1$ dan $S2$ }
<i>setDifference</i> : $S1 \times S2 \rightarrow S$	{ Menghasilkan $S1$ dikurangi $S2$ }
<i>copy</i> : $S \rightarrow S$	{ Mengcopy set S ke set baru }
<i>isSubset</i> : $S1 \times S2 \rightarrow \text{boolean}$	{ Mengembalikan true jika $S1$ adalah subset dari $S2$ }

Axiomatic Semantics (fungsional)

- 1) $\text{new}()$ returns a set
- 2) $\text{isIn}(v, \text{new}()) = \text{false}$
- 3) $\text{isIn}(v, \text{add}(v, S)) = \text{true}$
- 4) $\text{isIn}(v, \text{add}(u, S)) = \text{isIn}(v, S)$ if $v \neq u$
- 5) $\text{remove}(v, \text{new}()) = \text{new}()$
- 6) $\text{remove}(v, \text{add}(v, S)) = \text{remove}(v, S)$
- 7) $\text{remove}(v, \text{add}(u, S)) = \text{add}(u, \text{remove}(v, S))$ if $v \neq u$
- 8) $\text{isEmpty}(\text{new}()) = \text{true}$
- 9) $\text{isEmpty}(\text{add}(v, S)) = \text{false}$

Dengan S adalah set dan u, v adalah elemen.

(Dapatkah Anda membuat aksioma tambahan untuk union, intersection, setDifference?)

Implementasi Set dengan Tabel

Memori tempat penyimpan elemen adalah sebuah array dengan indeks $0..CAPACITY-1$.

Pertimbangan implementasi: operasi terhadap set biasanya memeriksa keanggotaan sebuah elemen di dalam set.

- Jika himpunan elemen set memiliki definisi urutan, elemen-elemen set dapat disusun secara terurut.
 - Efisiensi operasi pemeriksaan keanggotaan (`isIn`) \rightarrow algoritma *binary search*.
- Jika elemen-elemen set berukuran besar/berstruktur kompleks, operasi pemeriksaan keanggotaan (`isIn`) dapat menjadi sangat “mahal”.
 - Untuk efisiensi, bisa memanfaatkan *hash table*.

ADT Set (dengan array eksplisit-statik)

KAMUS UMUM

constant CAPACITY: integer = ... { *Banyaknya elemen maksimum* }

type ElType: integer { *elemen set* }

type Set:

 < buffer: array [0..CAPACITY-1] of ElType, { *array penyimpan elemen set* }
 length: integer { *jumlah elemen set* } >

Latihan

Tuliskan algoritma-algoritma untuk **add**, **remove**, **isIn**, dan **union** terhadap Set of integer, jika elemen-elemen Set disimpan secara:

1. Terurut berdasarkan nilainya

$S =$

7	12	26	32	47			
0	1	2	3	4	5	6	7

$\text{add}(S, 22)$

$S =$

7	12	22	26	32	47		
0	1	2	3	4	5	6	7

2. Berdasarkan urutan dilakukannya Insert

$S =$

12	26	7	47	32			
0	1	2	3	4	5	6	7

$\text{add}(S, 22)$

$S =$

12	26	7	47	32	22		
0	1	2	3	4	5	6	7

Lakukan analisis terhadap kedua alternatif cara penyimpanan, yang mana yang memberikan kinerja lebih baik untuk operasi-operasi tersebut!

Penjelasan Latihan: Ilustrasi Operasi

$s = \{12, 26, 7, 47, 32\}$

$t = \{5, 12\}$

$\text{add}(s, 22) \rightarrow s = \{12, 26, 7, 47, 32, 22\}$

$\text{add}(s, 26) \rightarrow s = \{12, 26, 7, 47, 32, 22\}$

$\text{remove}(s, 47) \rightarrow s = \{12, 26, 7, 32, 22\}$

$\text{isIn}(s, 7) = \text{true}$

$\text{isIn}(t, 7) = \text{false}$

$\text{union}(s, t) = \{12, 26, 7, 32, 22, 5\}$