Vectors

Christophe Rhodes

Motivation

- useful abstraction of memory
- basic building block

Definition

A vector is a finite fixed-size sequential collection of data.

finite	a vector has a non-negative integer length;
fixed-size	the length of the vector is immutable;
sequential	a vector has a defined and static storage order of its elements;
collection	a vector's contents represents data in the collection; things not stored in the vector are not in the collection.

Operations

length	return the number of elements in the vector
select[k]	return the k^{th} element of the vector
store![o,k]	set the k^{th} element of the vector to o

In pseudocode, respectively:

length	LENGTH(v)
select[k]	v[k]
store![o,k]	v[k] ← o

Constructor:

- **new** Vector(n)

Not Vector operations

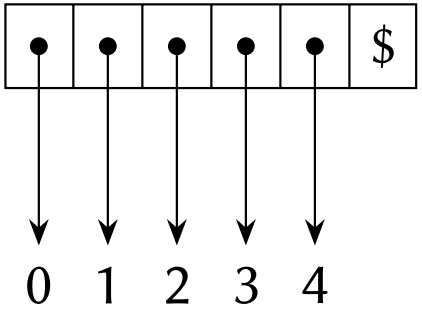
delete!

insert!

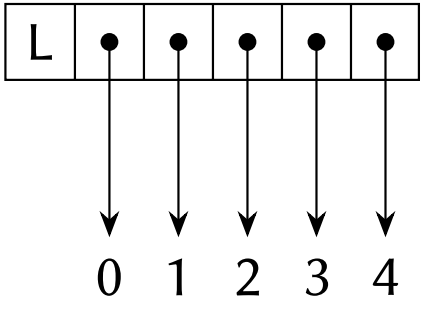
resize!

Implementation

sentinel (C strings)



length-data (everything else)



Complexity analysis

How much work do operations take?

select, store!

1. offset calculation: base address + k × element size
2. pointer read (select) or write (store!)

length

Depends on implementation strategy:

sentinel

1. initialize count to 0, position to 0
2. iterate position through string, incrementing count, until the sentinel ($)
3. return count

length-data

1. read the length slot

Work

1. reading
   - CLRS 10.3
   - Poul-Henning Kamp, “The most expensive one-byte mistake”, ACM Queue 9:7, 2011