Disjoint sets

CSCE 221

Data structure ADT - description

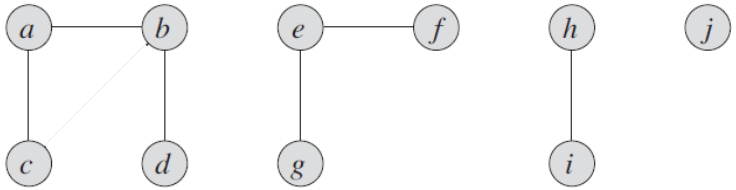
- A Disjoint-set data structure is a collection S={S1,....,Sk} of disjoint dynamic sets.

- Each disjoint set is identified by its representative

Data structure ADT - operations

- Make-Set(x)
  - Creates a single-element set whose only member and representative is x.
- Union(x, y)
  - Unites the sets containing x and y. The two original sets are removed from the sets collection.
- Find-Set(x)
  - Returns a pointer to the representative of the set containing x.

Example - Connected components



(a)

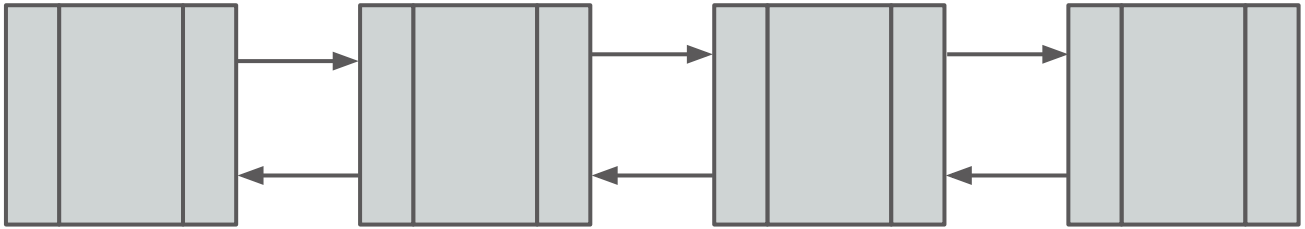
Edge processed	Collection of disjoint sets									
initial sets	{a}	{b}	{c}	{d}	{e}	{f}	{g}	{h}	{i}	{j}
(b,d)	{a}	{b,d}	{c}		{e}	{f}	{g}	{h}	{i}	{j}
(e,g)	{a}	{b,d}	{c}		{e,g}	{f}		{h}	{i}	{j}
(a,c)	{a,c}	{b,d}			{e,g}	{f}		{h}	{i}	{j}
(h,i)	{a,c}	{b,d}			{e,g}	{f}		{h,i}		{j}
(a,b)	{a,b,c,d}				{e,g}	{f}		{h,i}		{j}
(e,f)	{a,b,c,d}				{e,f,g}			{h,i}		{j}

Implementation - Linked Lists

- Each list is one of the disjoint sets
- Each node is a element on a set
- Node has the following pointers
  - Next, Previous

Implementation - Linked Lists

- Set representation



Implementation - Runtime analysis

- Not so efficient

- Make-Set
  - O(1)
- Find-Set
  - O(n)
- Union
  - O(n)

Implementation - Runtime analysis

- Improving the performance
  - Path compression
    - Each node should have a direct pointer to the representative of the Set.
  - Union by rank
    - Always append the shorter list to the longer. Fewer representative pointers to update.

Implementation - Runtime analysis

- With these two heuristics

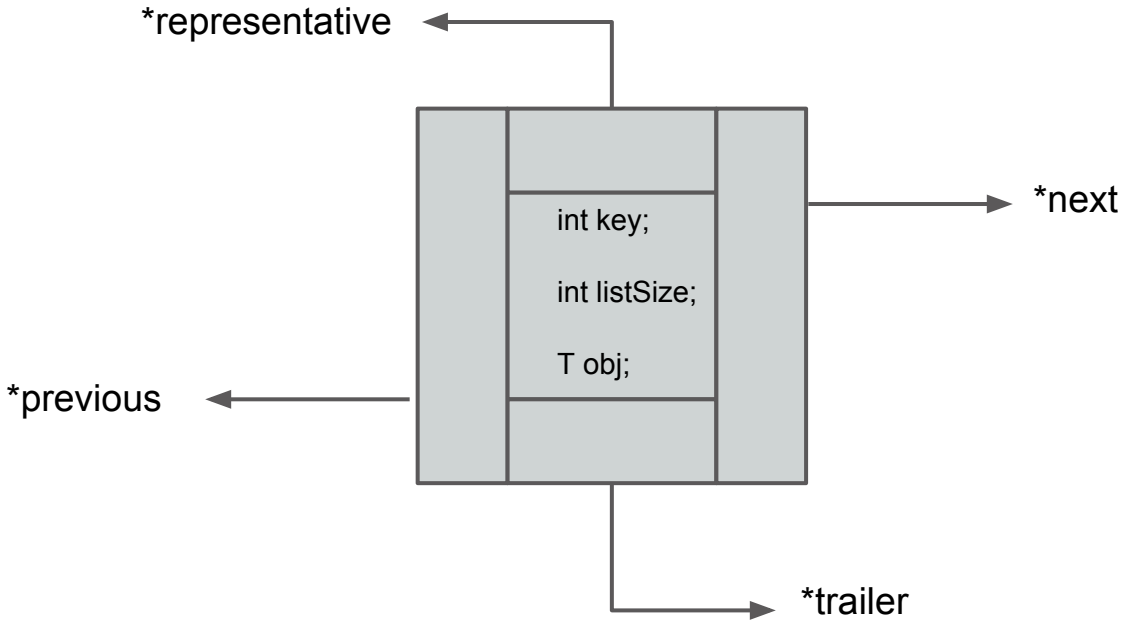
- Make-Set
  - O(1)
- Find-Set
  - O(1)
- Union
  - O(log n)

Implementation - Linked Lists

- Using the heuristics
  - Each node has the following pointers
    - Next, Previous
    - Representative - points to the head of the list
  - The representative node also has a pointer to the last element of the list.

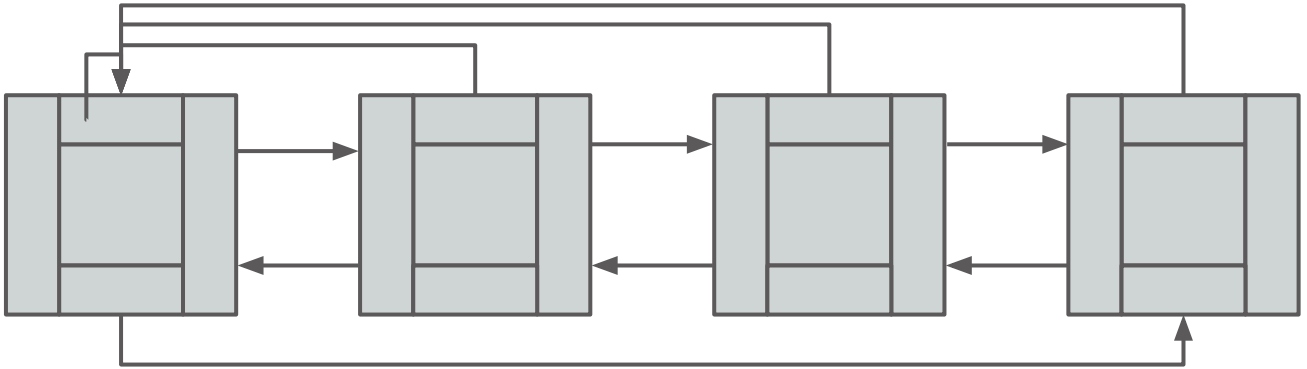
Implementation - Linked Lists

- Node representation



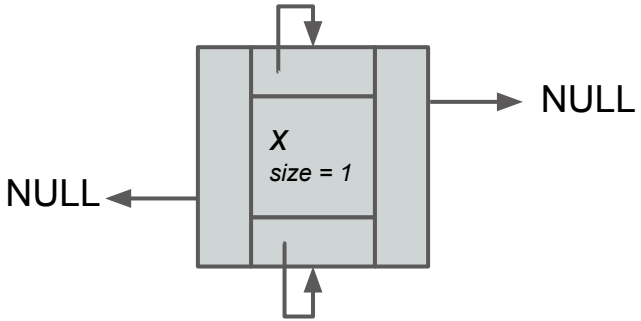
Implementation - Linked Lists

- Set representation



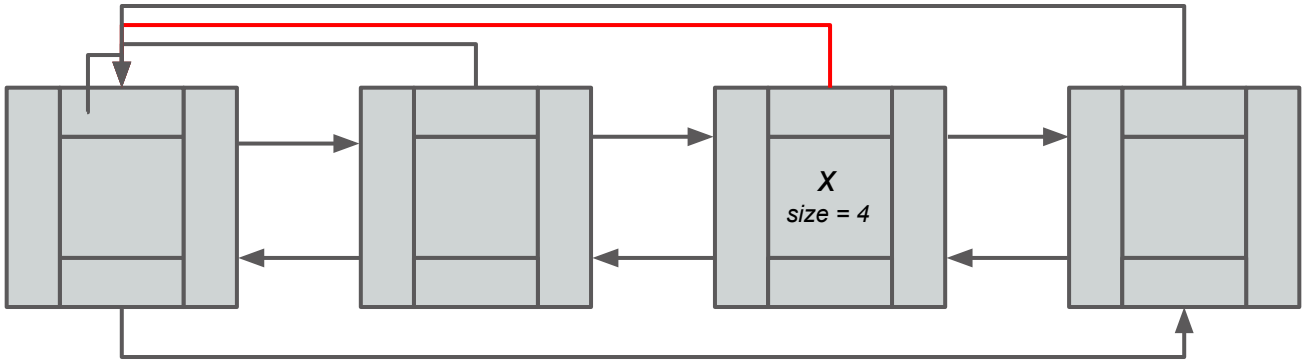
Implementation - Linked Lists

- Make-Set(x)



Implementation - Linked Lists

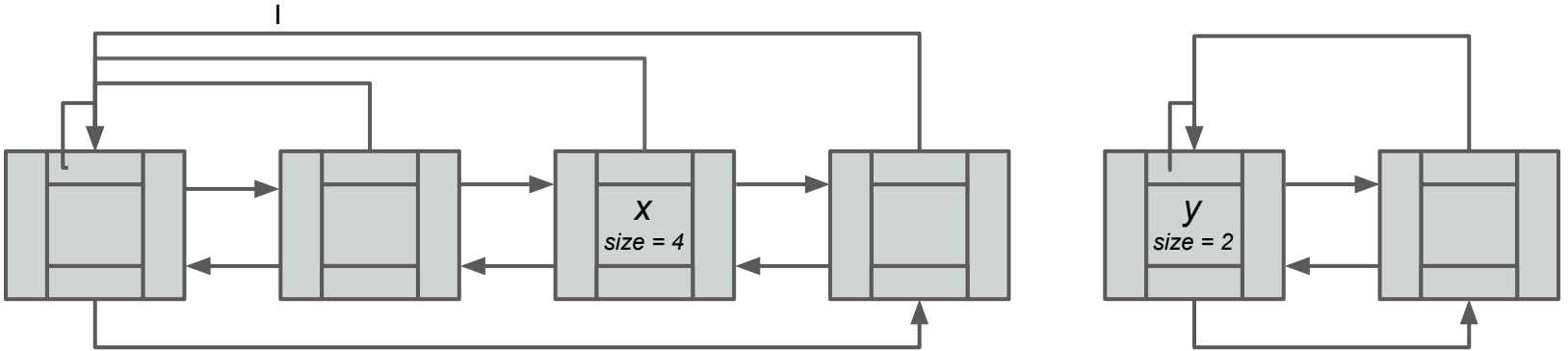
- Find-Set(*x*)



- Returns the node pointed by the representative pointer of node *x.*

Implementation - Linked Lists

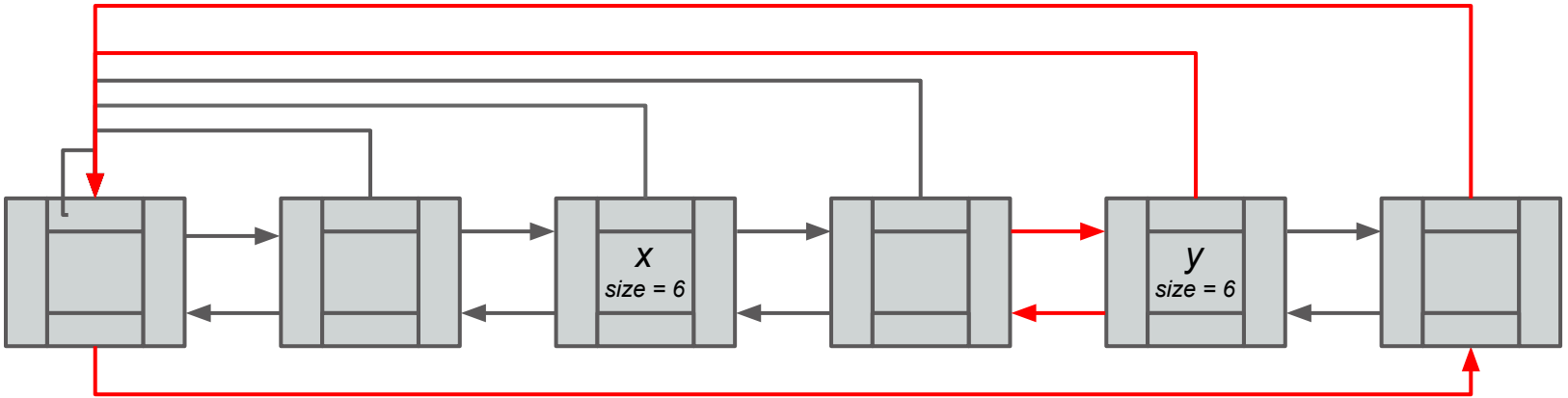
- Union(*x*, *y*)



- Append the shorter list on the longer.

Implementation - Linked Lists

- Union(*x*, *y*)



- Append the shorter list on the longer.