Linked Lists ctd..

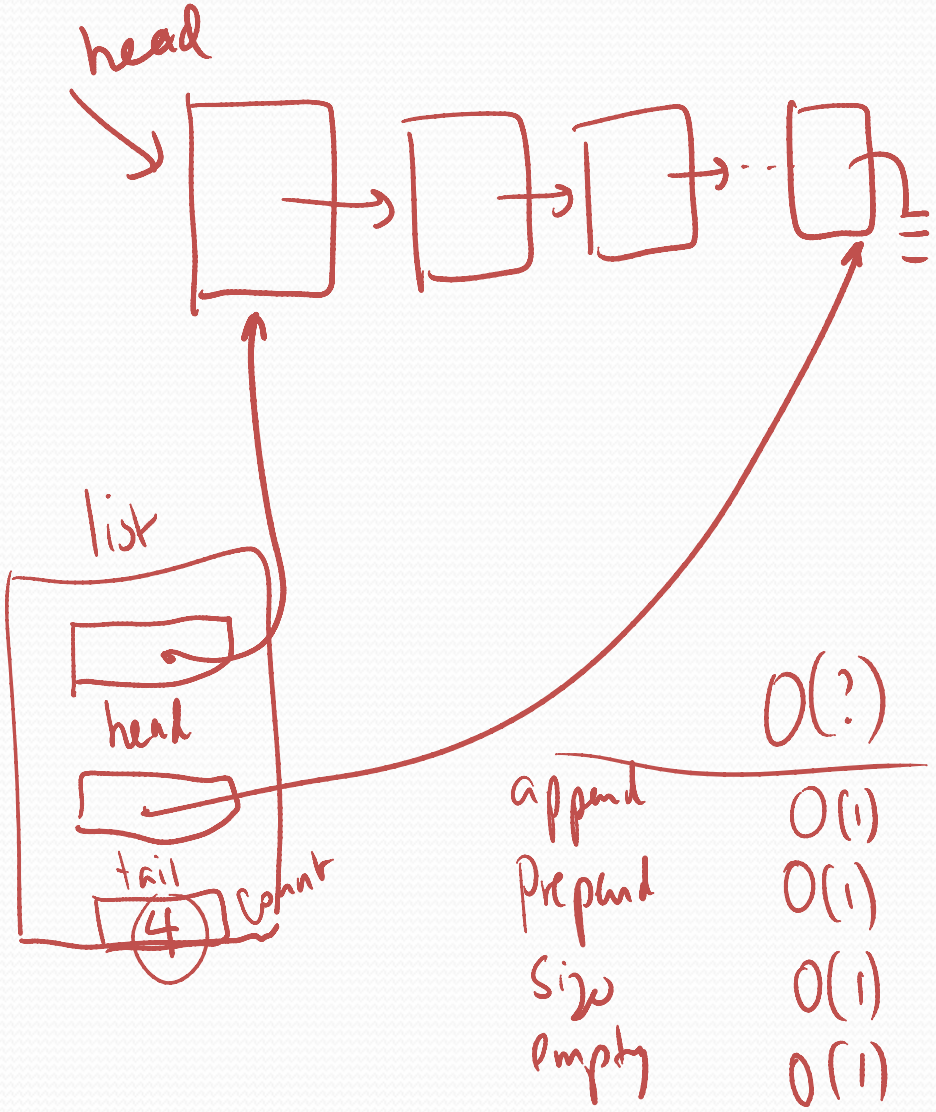
15-123

Systems Skills in C and Unix

Linked Lists wrapping

```
typedef struct node{
    void* data;
     struct node* next;
} node;

typedef struct list{
    node* head, *tail;
     int count;
} ;
```



Checking the integrity of a LL

```
int  isSegment(node* start, node* finish){
        if (start==finish) return 1;
        while (start != finish) {
                if (start->next == NULL) return 0;
                start = start->next;
        }
        return 1;
}
```

Reversing a list in O(n)?

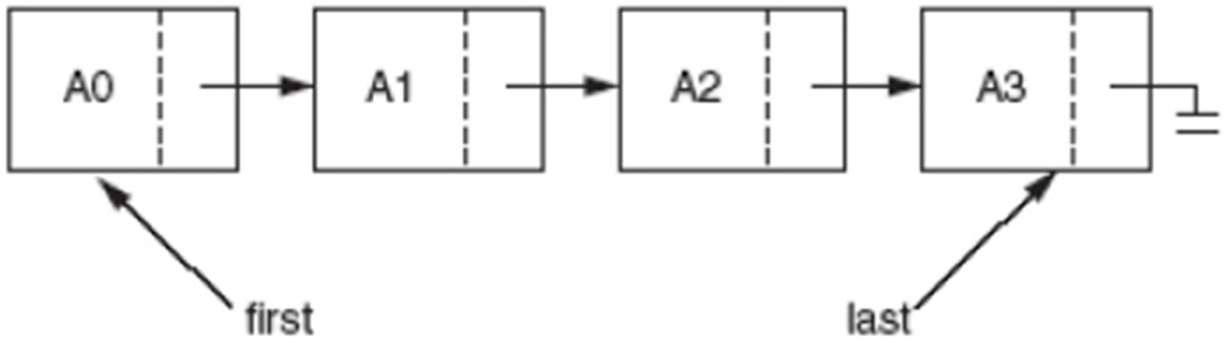
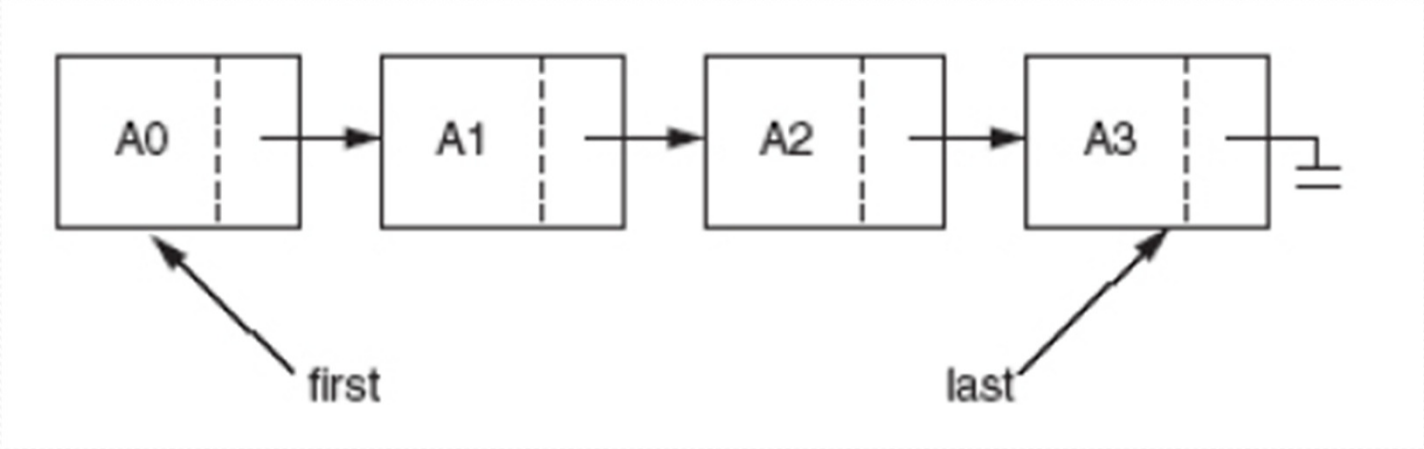
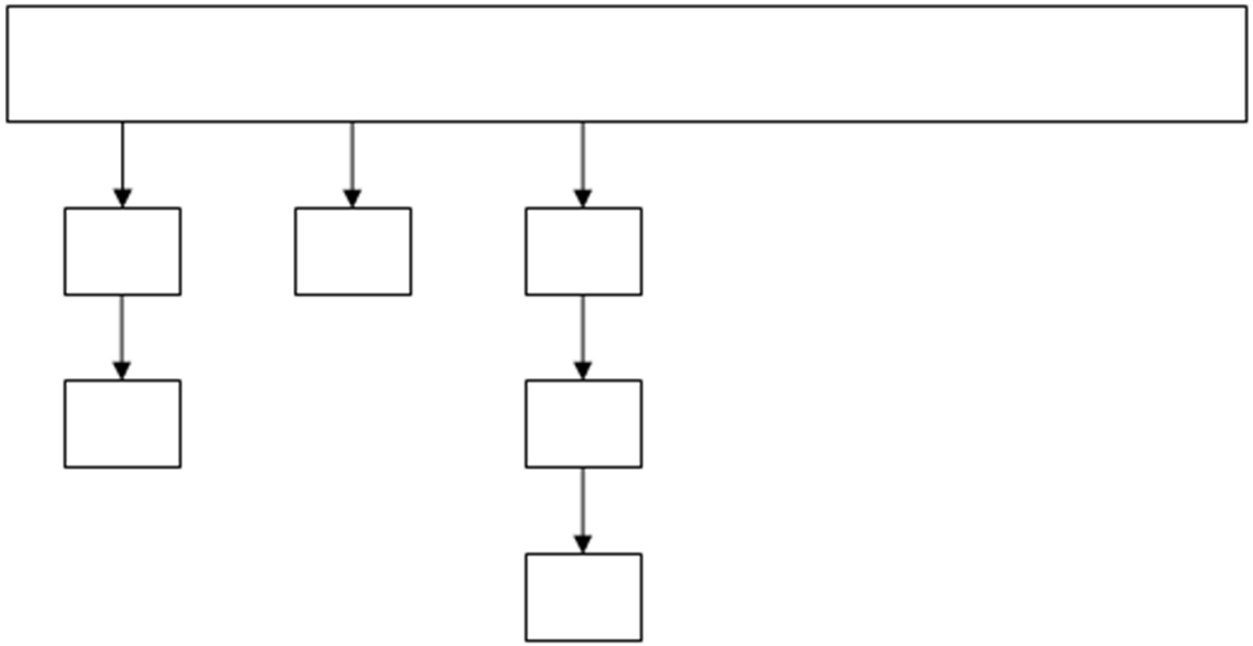


image source: Weiss Data Structures

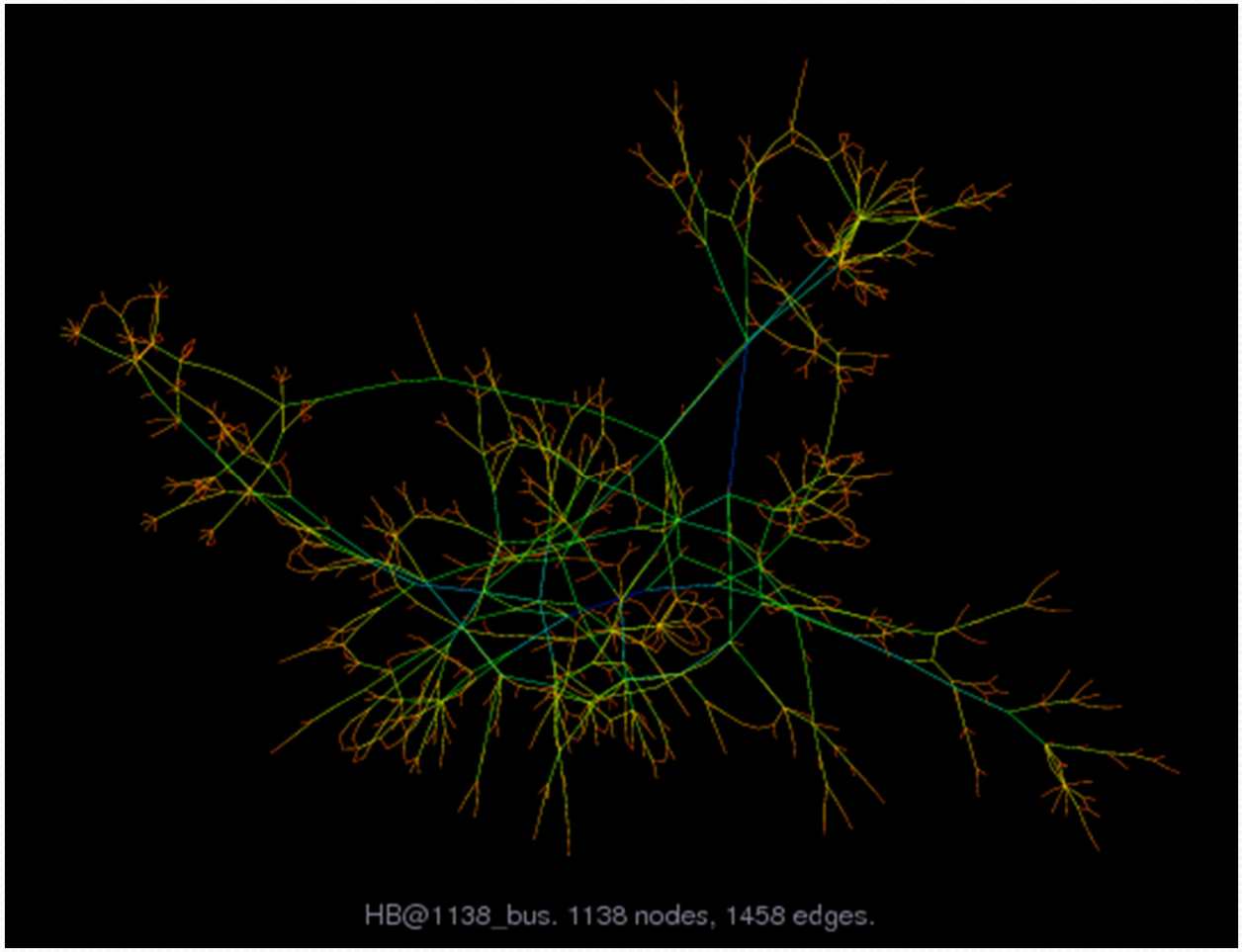
Insert in order



An Array of Linked Lists

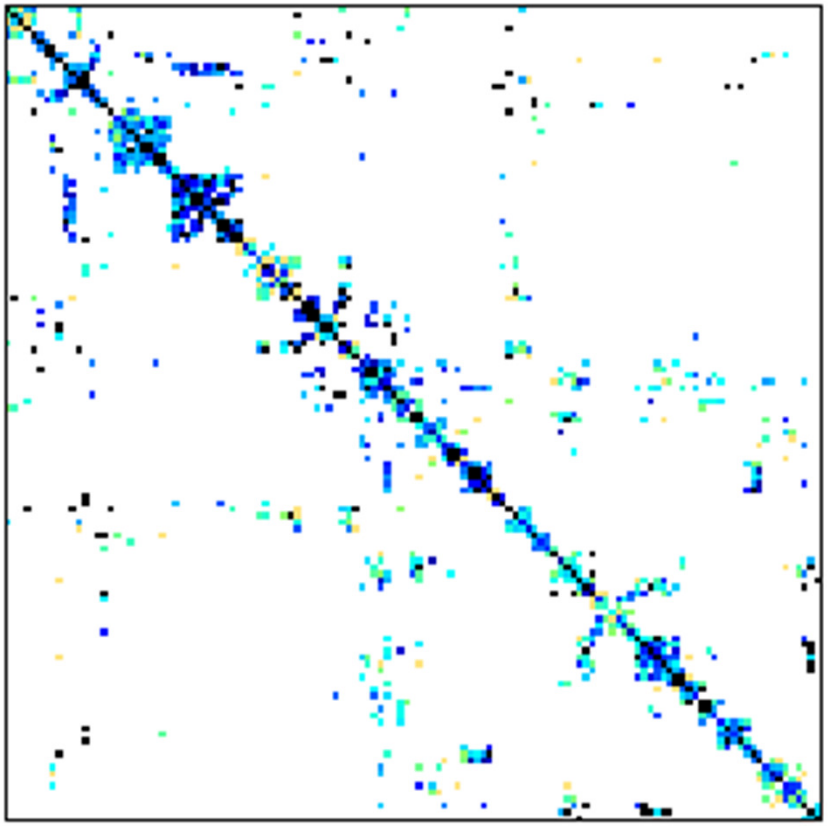


Sparse Matrices



Description: S ADMITTANCE MATRIX 1138 BUS POWER SYSTEM, D.J.TYLAVSKY, JULY 1985.

The matrix



Description: S ADMITTANCE MATRIX 1138 BUS POWER SYSTEM, D.J.TYLAVSKY, JULY 1985.

Storing Sparse Matrices

Structs used

```
typedef struct node {
        int row, column,
        double value;
        struct node*  rowPtr;
        struct node*  colPtr;
} node;
```

Structs used

```
typedef struct matrix {
        node**  rowList;
        node** columnList;
        int  rows, columns;
  } matrix;
```

Coding Examples