A list ADT

A list ADT:

```
template <class C>
class list{
public:
      list();
      bool isEmpty();
      bool isFull();
      void insert(C x, int p);
      C findkth(int p);
      void delete(int p);// delete the element in position p
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

A list ADT: implement as array

A list ADT:

```
template <class C>
class list{
public:
        list();                          O(1) time
        bool isEmpty();                  O(1) time
        bool isFull();                   O(1) time
        void insert(C x, int p);         O(n) time
        C findkth(int p);                O(1) time
        void delete(int p);// dele       O(n) time ement in position p
        int sizeof(); // how many elements in the list?
private:
        .
        .
};
```

A list ADT: implement as linked-list

A list ADT:

```
template <class C>
class list{
public:
        list();                                  O(1) time
        bool isEmpty();                          O(1) time
        bool isFull();                           O(1) time (just say "no")
        void insert(C x, int p);                 O(n) time
        C findkth(int p);                        O(n) time
        void delete(int p);// dele                O(n) time  ition p
        int sizeof(); // how many elements in the list?
private:
        .
        .
};
```

A modified list ADT

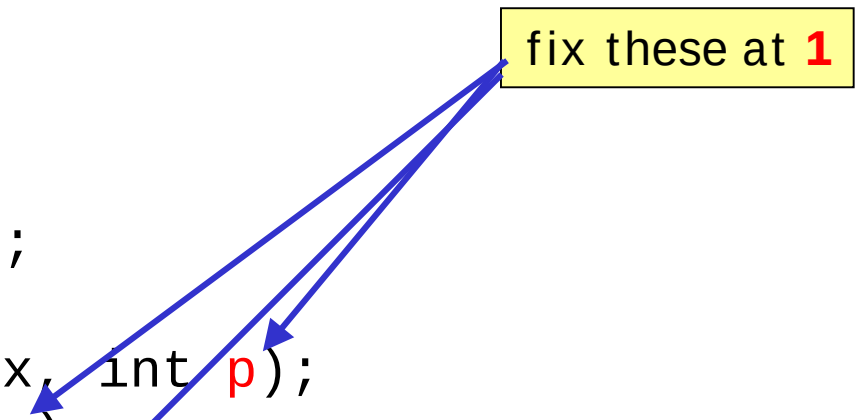
A list ADT:

```
template <class C>
class list{
public:
      list();
      bool isEmpty();
      bool isFull();
      void insert(C x, int p);
      C findkth(int p);
      void delete(int p);// delete the element in position p
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

A modified list ADT

A list ADT:

```
template <class C>
class list{
public:
        list();
        bool isEmpty();
        bool isFull();
        void insert(C x, int p);
        C findkth(int p);
        void delete(int p);// delete the element in position p
        int sizeof(); // how many elements in the list?
private:
        .
        .
};
```



A STACK ADT

A stack ADT:

```
template <class C>
class list{
public:
      list();
      bool isEmpty();
      bool isFull();
      void insert(C x, int 1);
      C findkth(int 1);
      void delete(int 1);// delete the element in position 1
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

A STACK ADT

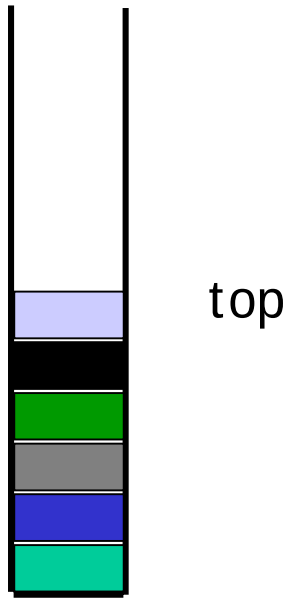
A stack ADT:

```
template <class C>
class stack{
public:
      stack();
      bool isEmpty();
      bool isFull();
      void push(C x);
      C top();
      void pop();// delete the element in position 1
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

Stacks

```
template <class C>
class stack{
public:
      stack();
      bool isEmpty();
      bool isFull();
      void push(C x);
      C top();
      void pop();
      int sizeof();
private:
      .
      .
};
```

```
stack<int> S1;
S1.push(5);
S1.push(10);
S1.push(8);
cout << S1.top();
S1.pop();
cout << S1.sizeof();
```



A modified list ADT

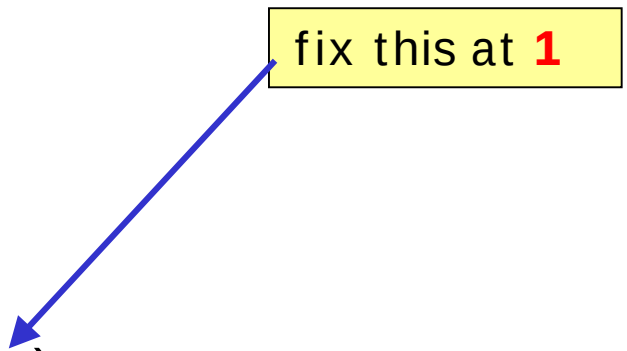
A list ADT:

```
template <class C>
class list{
public:
      list();
      bool isEmpty();
      bool isFull();
      void insert(C x, int p);
      C findkth(int p);
      void delete(int p);
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

A modified list ADT

A list ADT:

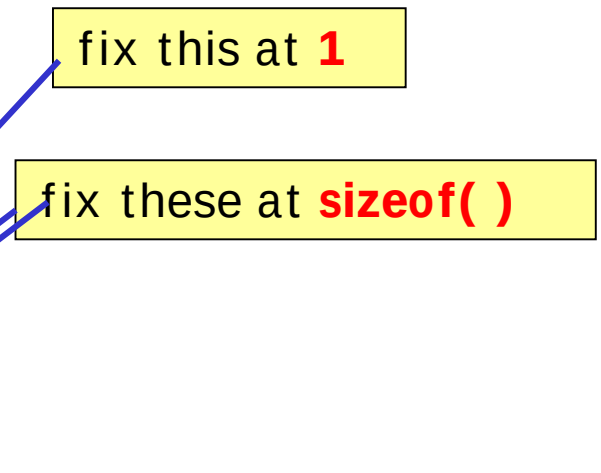
```
template <class C>
class list{
public:
        list();
        bool isEmpty();
        bool isFull();
        void insert(C x, int p);
        C findkth(int p);
        void delete(int p);
        int sizeof(); // how many elements in the list?
private:
        .
        .
};
```



A modified list ADT

A list ADT:

```
template <class C>
class list{
public:
        list();
        bool isEmpty();
        bool isFull();
        void insert(C x, int p);
        C findkth(int p);
        void delete(int p);
        int sizeof(); // how many elements in the list?
private:
        .
        .
};
```



A QUEUE ADT

A stack ADT:

```
template <class C>
class list{
public:
      list();
      bool isEmpty();
      bool isFull();
      void insert(C x, int 1);
      C findkth(int sizeof());
      void delete(int sizeof());
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

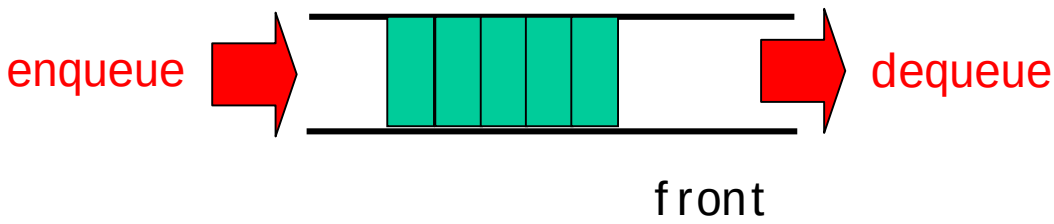
A QUEUE ADT

A queue ADT:

```
template <class C>
class queue{
public:
      queue();
      bool isEmpty();
      bool isFull();
      void enqueue(C x);
      C front();
      void dequeue();
      int sizeof(); // how many elements in the list?
private:
      .
      .
};
```

Queues

```
template <class C>
class queue{
public:
      queue();
      bool isEmpty();
      bool isFull();
      void enqueue(C x);
      C front();
      void dequeue();
      int sizeof();
private:
      .
      .
};
```




```
queue<int> Q1;
Q1.enqueue(5);
Q1.enqueue(10);
Q1.enqueue(8);
cout << Q1.front();
Q1.dequeue();
cout << Q1.sizeof();
```