CSC215

Math and Computer Science

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

An Unordered List (think of a todo list)

- Create a list that contains strings
- User can insert anywhere within the list
- User can remove from anywhere within the list

- Private Data
  - Array of strings
  - How many items are in the list

Data Abstraction

Create

Destroy

Insert an item

Remove an item

Find an item

Retrieve an item

See if list is empty

Count the number of items

Print the list

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Functional Abstraction

- Create – create an empty list

  In: nothing out: nothing

- Destroy

  In: nothing out: nothing

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Functional Abstraction

Insert – place an item at a given location in the list

in: item, position out: T/F flag for success

Remove – remove an item from the list with a particular value

in: item out: T/F flag for success

Remove – remove an item from the list at a given location

in: position out: T/F flag for success

Functional Abstraction

- Retrieve an item at a particular location

in: position	out: T/F flag and item

- Find an item in the list

in: item	out:	T/F flag and position
	or	position, -1 Not Found

Functional Abstraction

- See if list is empty

  in: nothing | out: T/F flag

- Retrieve number of items in list

  in: nothing | out: count of items in list

- Print the list

  in: ostream | out: nothing

The Unordered List Definition (.h file)

```
class unorderedList
{   public:
        unorderedList();
        ~unorderedList();

        bool insert( char *item,
                     int pos );
        bool remove( char *item );
        bool remove( int pos );

        bool retrieve( char *item,
                       int pos );
        int find( char *item );

        bool isEmpty();
        int size();
        void print( ostream &out );

    private:
        char theList[20][30];
        int numItems;
};
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

In Depth Look at the Functions

- Create

  Will set the variable numItems to zero.

- Destroy

  Will do nothing. No Dynamic memory

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

In Depth Look at the Functions

- Insert
  - Will Check the value of position to make sure it is between the range of 0 to numItems. numItems would be the first open spot in the list
  - Will make sure that there is room to store the item
  - If the item name is more than 29 characters, truncation will happen when copying it over.
  - If the position to be inserted is in the middle of the list, we need to move all items up 1 spot
  - numItems will be incremented upon successful insert.

Before:	Grocery	Eval	Grade	Laundry	
After:	Grocery	Eval	Room for newitem	Grade	Laundry

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

In Depth Look at the Functions

- Remove
  - Pass in an item
  - Will move through the data and remove the first instance of it when found.
  - If not found, a false will be returned otherwise a true will be returned
  - When found, data will be moved down to remove it from the list
  - NumItems will be decremented if successful

Before:	Grocery	Eval	Grade	Laundry	
After:	Grocery	Eval	Laundry		

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

In Depth Look at the Functions

- Remove
  - Pass in a position
  - Will make sure the position is a valid spot in the data
  - Data will be moved down to remove it from the list
  - numItems will be decremented if successful
  - A True or False will be returned based on if it succeeded.

Before:	Grocery	Eval	Grade	Laundry	
After:	Grocery	Grade	Laundry		

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

In Depth Look at the Functions

- Retrieve
  - Given a position copy the item at the spot into a character array.
  - Validate that position is within the range of valid data
  - Return a true if you copied an item into the char array, false if you didn't.

- Find
  - Given an item, return the position of that item.
  - If the item is not found, return a -1.

In Depth Look at the Functions

- Is Empty

  Will return True if the list is empty false otherwise

- Size

  Will return the number of Items in the list

- Print

  Will output all items in the list to a given ostream.

SOUTH DAKOTA
SCHOOL OF MINES & TECHNOLOGY

Position

- Will 0 be the first spot
- Will 1 be the first spot
- Need to Decide now before we write the code.
  - I choose that 1 will be the first spot.
  - Rethink the find function now
  - Will return 0 if not found rather than a -1

Which Don't Change the Data

- Retrieve
- Find
- Is Empty
- Size
- Print

- Add the word const after the function prototypes and function to add additional protection

Definition with Const

```
class unorderedList
{   public:
        unorderedList();
        ~unorderedList();

        bool insert( char *item,
                     int pos );
        bool remove( char *item );
        bool remove( int pos );
        bool retrieve( char *item,int pos ) const;
        int find( char *item ) const;
        bool isEmpty() const;
        int size() const;
        void print( ostream &out )const;

    private:
        char theList[20][30];
        int numItems;
};
```

Making Some Defaults

- Default positions, must be done right to left
- Insert, make position 1 if not given
    - Mylist.insert("Buy Groceries");
    - Mylist.insert("Buy Groceries" , 2);
- Remove, make position 1 if not given
    - Mylist.remove( );
    - Mylist.remove(3);
- Retrieve, make position 1 if not given
    - Mylist.retrieve(item);
    - Mylist.retrieve(item, 4);

New Prototypes for Definition

```
bool insert( char *item, int pos=1 );
bool remove( int pos=1 );
bool retrieve( char *item, int pos=1 ) const;
```

Refined Definition

```
class unorderedList
{   public:
        unorderedList();
        ~unorderedList();

        bool insert( char *item,
                     int pos=1 );
        bool remove( char *item );
        bool remove( int pos=1 );
```

```
bool retrieve( char *item, int pos=1)const;
int find( char *item ) const;
bool isEmpty() const;
int size() const;
void print( ostream &out ) const;

    private:
        char theList[20][30];
        int numItems;
};
```

SOUTH DAKOTA
SCHOOL OF MINES
& TECHNOLOGY