CSC 215

Math and Computer Science

SOUTH DAKOTA
SCHOOL OF MINES
& TECHNOLOGY

Algorithms

- Library that contains a lot of useful functions
- Work with iterators and pointers
- Only a sample of what is available
- Visit: C PLUS PLUS
- #include <algorithm>

Transform Function

- Most useful function available
- Will walk through the iterator range applying some function to each element.
- Must have space already reserved for the transform function

Transform Examples

```
string str = "Hello";
string dest;

transform( str.begin(), str.end(), str.begin(), toupper);
// str = "HELLO"

dest.resize( str.size() );
transform( str.begin(), str.end(), dest.begin(), toupper);
// str = "Hello", dest = "HELLO"
```

SOUTH DAKOTA
SCHOOL OF MINES
& TECHNOLOGY

Transform Examples

```
char str[100] = "Hello";
char dest[100];
transform( str, str+strlen(str), dest, toupper);
dest[strlen(str)] = '\0';
// str = "Hello", dest = "HELLO"
```

SOUTH DAKOTA
SCHOOL OF MINES
& TECHNOLOGY

Transform Examples – Your Own Function

- Prototype for function is:

```
datatype funciontname( datatype item );
```

```
transform(str.begin(), str.end(), str.begin(), passwd);
char passwd( char ch )
{   if( ch == 'e' || ch == 'E' )
        return '3';
    if( ch == 'l' || ch == 'L' )
        return '1';
    return ch;
}
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Reverse Function

- Reverse function flips the contents between begin and end

```
string str = "Hello";
char str2[100] = "GoodBye";

reverse(str.begin(), str.end());
reverse( str2, str2+strlen(str2));
// str = olleH
// str2 = eyBdooG
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Reverse Examples

```
string str = "Hello";
char str2[100] = "GoodBye";

reverse(str.begin()+1, str.end()-1);
reverse( str2, str2+4);

// str = Hlleo
// str2 = dooGBye
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Sort Function

- Sort the data between the two iterators
- Default is increasing order
- Can write a function to do your own comparison

```
bool functionName( datatype lhs, datatype rhs );
```

Sort Example

```
string str = "Roger Schrader";
char str2[100] = "Roger Schrader";

sort( str.begin(), str.end() );
sort( str2, str2 + strlen(str2) );
// str = RSacdeeghorrr
// str2 = RSacdeeghorrr
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Sort Examples

```
string str = "Roger Schrader";
char str2[100] = "Roger Schrader";

sort( str.begin(), str.end()-9 );
sort( str2+6, str2 + strlen( str2 ) );
// str  = Regor Schrader
// str2 = Roger Sacdehrr
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Count Function

- Sum the number of items that match a particular value between two iterators.

```
string str = "Roger Schrader";
char str2[100] = "Roger Schrader";
int num, num2;
num = count( str.begin(), str.end(), 'e' );
num2 = count( str2, str2 + strlen( str2 ), 'r' );
// num = 2, num2 = 3
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

Count If Function

- Sums up the number of data items that match a particular condition.
- You must supply a function with the following syntax:

```
bool   functionName ( datatype  value );
```

  - This function returns true if the value is to be counted
  - False if the value is not to be counted.

Count If Example

```
string str = "Roger Schrader";
char str2[100] = "Roger Schrader";
int num, num2;

num = count_if( str.begin(), str.end(), isVowel);
num2 = count_if( str2, str2 + strlen( str2 ),
                 isVowel );
// num = 4, num2 = 4
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY

isVowel Function

```
bool isVowel( char ch )
{
    if( ch == 'a' || ch == 'A' || ch == 'e' ||
        ch == 'E' || ch == 'i' || ch == 'I' ||
        ch == 'o' || ch == 'O' ||
        ch == 'u' || ch == 'U' )
        return true;
    return false;
}
```

SOUTH DAKOTA SCHOOL OF MINES & TECHNOLOGY