Intermediate/Advanced Computer Programming
Announcements

Quiz Friday (July 7\textsuperscript{th})
- Approximately 30 minutes
Review

• What is the difference between a pointer and a reference?
  – How do we define pointers?
    • What operators are used in conjunction with pointers?
  – How do we define references?

• What is the difference between member and non-member functions?

• Describe function overloading

• How can we overload an operator in a class?

• What is a conversion constructor?

• What are the different contexts where the const keyword can be used?
Private data can be accessed by member functions of a class/object
Private members

Accelerate
Brake
Change gear
Accessing private data

Private data can be accessed by member functions of a class/object.

What if we want to allow a non-member function access to private data?

- Make the non-member function a friend.
- `friend` functions have access to private member functions/data of the class that declares them a friend.
Private members

- Accelerate
- Brake
- Change gear

Gear
Fuel flow rate
Speed

friend
Who needs friends?

• May not have access to the source code of the class
  – Example: ostream class

• Some operator overloads have more flexibility when declared as friends
  – Example:

```cpp
BigInt a(“34”);
9 + a
```
Comparison Operators

Used to compare two objects

bool operator==(const X& lhs, const X& rhs)
bool operator!=(const X& lhs, const X& rhs)
bool operator<(const X& lhs, const X& rhs)
bool operator<=(const X& lhs, const X& rhs)
bool operator>(const X& lhs, const X& rhs)
bool operator>=(const X& lhs, const X& rhs)
Comparison Operators

standard library’s algorithms (e.g. std::sort()) will always only expect operator< to be present

Can we implement all comparisons by defining operator< and using it to implement all other comparison operators?
Counting/Enumerating
Counting/Enumerating

... the number of LEDS that should be illuminated ...

... number of possible combinations ...

... count of the number of substrings ...
Counting/Enumerating

Sometimes it is more efficient than looping

Useful to check output/verify iteration over all possibilities

Provides a way to quantify the maximum number of basic operations

- Gives us a handle on the complexity of a given algorithm
- Can determine whether the program will finish in a few seconds or a few weeks
Counting/Enumerating

Longest common substring in two BigInts

- Substring is a subset of the string, where the order of the elements in the subset is preserved

Implement two functions

- ListSubstrings()
  - Enumerate all possible substrings of one BigInt
- LCS()
  - Find the longest common subsequence of two BigInts
make and Makefiles