

Lec 14 Classes

Operator Basics

- You can define almost all C++ operators for class, this is called **operator overloading**.
- You can define your own `+`, `-`, `*`, `/`, `%`, `[]`, `()`, `^`, `!`, `<<`, `<=`, `<`, ...
- We use it when we want to provide conventional notation for a type we design.

Thing to know

- An overload operator must have at least one user-defined type.
- `int operator+(int, int)` // error, you can't overload built-in `+`
- Define operator with their conventional meaning

Overload operator <<

- The overload operator << takes two arguments, the first argument is ostream&, the second argument is the object.
- It must be implemented as non member function.

- Format

```
ostream& operator<<(ostream& os, const T& obj)
{
    // write obj to stream
    return os;
}
```


Implement a 2D vector class

- Similar to the graphics Point class
- a point is (x, y)
- need to have constructors, support for operators such as
 - $+$, $-$, dot (dot product), $/$ (point-wise division), $*$ (point-wise multiplication)