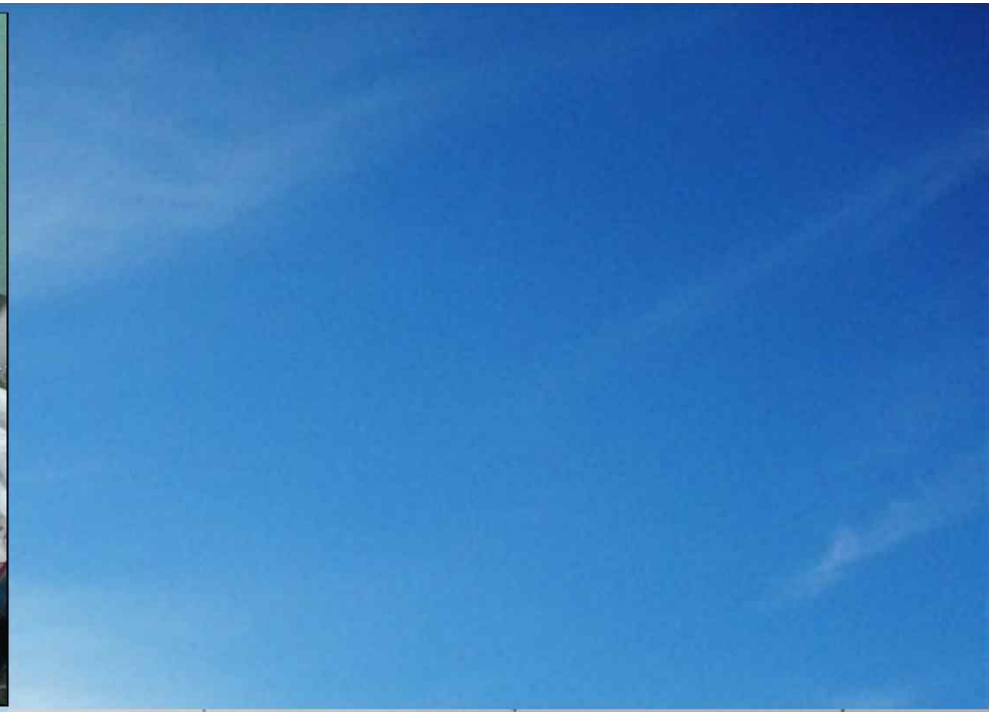
자료 구조
(Data Structures)

2018년 봄 학기
김태환 교수

병민아, 어디있니?

nae0a.com



This course is about…

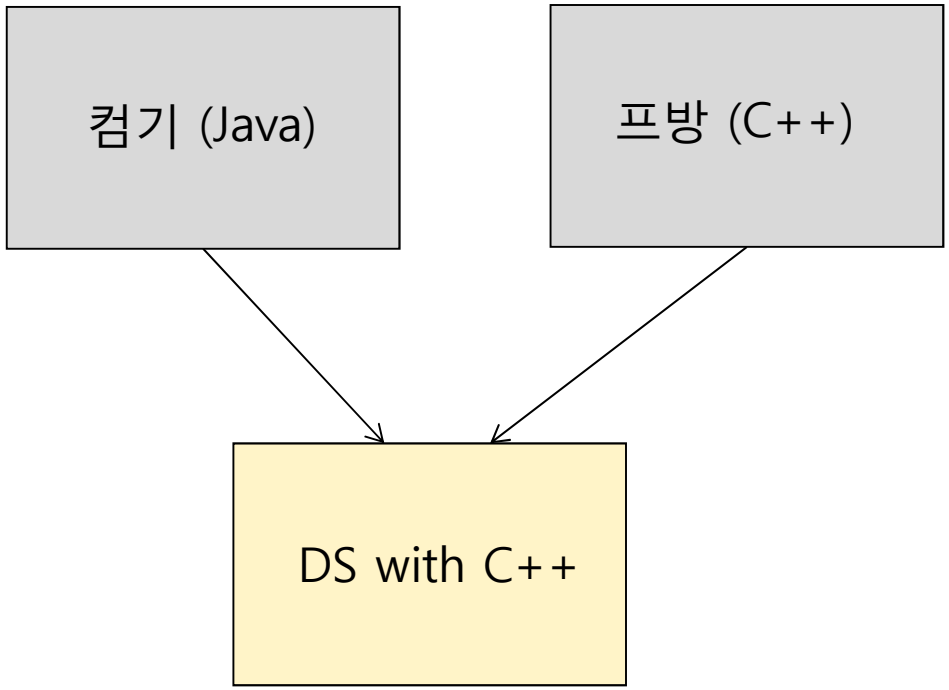
Data abstractions: elementary data structures (lists, stacks, queues, and trees) and their implementation using an object-oriented programming language; **Solutions to a variety of computational problems** such as search on graphs and trees; **Elementary analysis of algorithms**.

이 수업은....?

- A programming and “thinking” course
  – Not an easy course if you are “lazy”
  – Expect to work hard

- A fundamental computer science course
  – Must know if you claim to be a computer scientist
  – Must know if you want to be a good programmer and computer engineer
  – Essential for many follow up courses

Where did you come from?



Classes in C++:

Every variable is defined by _____, _____,

_____, _____

Primitive types:

```
int myInt;
char grade = 'A';
double t = 201.5;
```

User defined types:

```
sphere mySphere;
```

Class is a group of _____ and _____

Structure of a class defn:

How do we implement `sphere mySphere;` ?

```
class sphere {

  // member declarations

  ....




};
```

`Sphere` **member function definitions.**

Structure of a class defn (cont):

```
class sphere{

public:



private:



};
```

```
int main(){



}
```

`sphere` representation:

`sphere` functionality:

1. _____
2. _____
3. _____

`Sphere` member function definitions.

Structure of a class defn (cont):

```
class sphere{

public:
sphere();
sphere(double r);
void setRadius(double newR);
Double getDiameter() const;
…

private:
Double theRadius;

};
```

```
// constructors
  (next page)

void sphere::setRadius(double
newR) {

}

Double sphere::getDiameter()
const {

}
…
```

Differences with Java

1. _____
2. _____
3. _____

In our sphere class …

sphere.h

```
class sphere {

};
```

main.cpp

```
#include "sphere.h"

int main() {
  sphere a;

}
```

1. Does this code compile?
2. Does it run?

Access control and encapsulation:

sphere.h

```
class sphere {
    double theRadius;
};
```

main.cpp

```
#include "sphere.h"
#include <iostram>
using namespace std;

int main() {
  sphere a;
  cout << a.theRadius << endl;
}
```

1. Does this code compile?
2. Does it run?
3. In C++ class members are, by default, "private". Why would we want to hide our representation of an object from client?

Constructors: when you declare a sphere, a sphere class constructor is invoked.

```
…
// default constructor
sphere::sphere() {

   ___________________
}


// constructor with given radius
sphere::sphere(double r) {
     if ( r > 0)

        ____________
      else _______________
}
…
```

Remember about ctors:

1. _____
2. _____
3. _____

```
int main(){




}
```