

Java

C++

```
Person p(22, "Alice");
Person *p2 = new Person(21, "Bob");
...
delete p2;
```

```
Person p = new Person(...);
f(Person t) { }
```

```
f(Person *x) { x->grow(); }
g(Person *y) { y->grow(); }
h(Person &z) { z.grow(); }
```

```
f(p); // doesn't grow older
g(&p); // grows older
g(p2); // grows older
h(p); // only in C++, grows older
h(p2); // type mismatch
f(p2); // " " "
```

f is slowest

C++ STL

map (associative array)
lookup table, hash table
key → value

example: ID → name

state → population
key must be integer
key must be distinct

```
template <class key, class value>
class ECE462Map {
```

public:

```
ECE462Map() ECE462Map();
```

```
bool contains(key k);
```

```
void insert(key k, value v);
```

```
value get(key k);
```

```
void remove(key k);
```

```
int getsize();
```

```
virtual ~ECE462Map();
```

private:

```
key * keyArray;
```

```
value * valueArray;
```

```
int size; int item;
```

};