

C++ Primer

Jakub Marek

Handing over

Typecasts

Containe

Iterators

Templates

Programming in C++ - Primer Lesson 7 - Typecasts, Iterators & Templates

Jakub 'Eremiell' Marek <marekj14@fel.cvut.cz>

Silicon Hill C++ Academy

2013/12/02



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

Container

Iterators

Template

- 1 Revision
- 2 Handing over
- 3 Typecasts
- 4 Containers
- 5 Iterators
- 6 Templates



Welcome!

C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

_ .

. .

Template



C++ Primer

Jakub Marek

Revision

Handing a...

Typecasts



C++ Primer

Jakub Marek

Revision

.. ..

Typecasts

. .

Template

Declaration



C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

.

Templates

Declaration

■ header file



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

Container

Iterator

Template

Declaration

- header file
- included wherever we use it



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

iterators

Templates

Declaration

- header file
- included wherever we use it
- includes whatever is needed



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

31

Iterators

Template

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

7.

iterator:

Template:

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

source file



C++ Primer

Jakub Marel

Revision

Handing ove

Typecasts

31.

Iterators

Template:

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

- source file
- includes the header file



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

٠.

Iterator:

Template

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

- source file
- includes the header file
- is usually the only include



C++ Primer

Revision

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

- source file
- includes the header file
- is usually the only include

Access rights



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Iterators

Template

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

- source file
- includes the header file
- is usually the only include

Access rights

public



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

.

.

Template

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

- source file
- includes the header file
- is usually the only include

Access rights

- public
- protected



C++ Primer

Jakub Marel

Revision

Handing ove

- .

Typecasts

Containe

Iterator

.

Template

Declaration

- header file
- included wherever we use it
- includes whatever is needed

Definition

- source file
- includes the header file
- is usually the only include

Access rights

- public
- protected
- private



C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

Container

Template



C++ Primer

Jakub Marek

Revision

Handing ov

Typecasts

recrators

Template



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

~ . .

Template:

Encapsulation

helps data safety



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

Typecasts

Containe

Iterators

Template:

Encapsulation

- helps data safety
- lacksquare getters/setters



C++ Primer

Jakub Mare

Revision

Handing over

T.

Iterators

Tamanlaka

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism



C++ Primer

Jakub Marel

Revision

landing over

Typecasts

.

iterators

Template

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

■ related classes should inherit



C++ Primer

Jakub Mare

Revision

Handing ove

T. ... - - - - - -

.) pecuses

Iterators

Templates

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization



C++ Primer

Jakub Mare

Revision

Handing ove

T.

Typecasts

Containe

Iterators

Templates

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding



C++ Primer

Jakub Mare

Revision

Handing ove

Typocasts

.) pecuses

Iterators

Template:

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding

Operators



C++ Primer

Jakub Mare

Revision

Handing ove

T

Typecasis

Itaratar

Templates

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding

Operators

can be overloaded



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

Iterators

Template:

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding

Operators

- can be overloaded
- beware of member/non-member operators!



C++ Primer

Jakub Mare

Revision

Handing ove

T

Typecasts

Contain

l+ora+or

Template

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding

Operators

- can be overloaded
- beware of member/non-member operators!

Friendship



C++ Primer

Jakub Mare

Revision

Handing ove

T.....

Typecasis

Contain

l+ora+or

Templates

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding

Operators

- can be overloaded
- beware of member/non-member operators!

Friendship

breaks encapsulation



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

.) [-----

.

Iterators

Templates

Encapsulation

- helps data safety
- getters/setters

Inheritance & Polymorphism

- related classes should inherit
- specialization/generalization
- early/late binding

Operators

- can be overloaded
- beware of member/non-member operators!

Friendship

- breaks encapsulation
- helps divided abstractions



C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Iterators

Template



C++ Primer

Jakub Mare

Revisio

Handing over

Typecasts

Iterators

Template

once again three possibilities



C++ Primer

Jakub Mare

Revisio

Handing over

Typecasts

71

Iterators

Template

once again three possibilities

■ by value



C++ Primer

Jakub Marel

Revision

Handing over

Typocasts

.,,,.....

Containers

Iterator

once again three possibilities

- by value
- by pointer



C++ Primer

Jakub Mare

Revisio

Handing over

Typecasts

.) |- -----

Containers

Iterators

.

Template

once again three possibilities

- by value
- by pointer
- by reference



C++ Primer

Jakub Mare

D ... 1.1....

Handing over

Typecasts

Гурссазаз

Containe

Iterator

Template

once again three possibilities

- by value
- by pointer
- by reference

to pass by value means to copy



C++ Primer

Jakub Mare

Revisio

Handing over

Typecasts

.) pecuses

Container

.

Template

once again three possibilities

- by value
- by pointer
- by reference

to pass by value means to copy you've got to have



C++ Primer

Handing over

once again three possibilities

- by value
- by pointer
- by reference

to pass by value means to copy you've got to have

copy ctor



C++ Primer

Jakub Mare

Revisio

Handing over

Typecasts

31

Container

.

_ .

Template

once again three possibilities

- by value
- by pointer
- by reference

to pass by value means to copy you've got to have

- copy ctor
- copy assignment operator



C++ Primer

Jakub Mare

Revision

Handing over

T.

Typecasts

Container

Template

once again three possibilities

- by value
- by pointer
- by reference

to pass by value means to copy you've got to have

- copy ctor
- copy assignment operator
- dtor



C++ Primer

Jakub Mare

Revision

Handing over

T....

Typecasts

Containers

Iterators

Template

once again three possibilities

- by value
- by pointer
- by reference

to pass by value means to copy you've got to have

- copy ctor
- copy assignment operator
- dtor

The Rule of Three



C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Iterators

rempiate



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

Containor

.

iterators

Template

passing by pointers & references gives original



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

Iterators

.

Templates

passing by pointers & references gives original you can't change reference



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

.

Iterators

recrators

Templates

passing by pointers & references gives original you can't change reference you have to initialize reference at creation



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

Container

. .

iterators

Templates

passing by pointers & references gives original you can't change reference you have to initialize reference at creation reference cannot be NULL



Questions?

C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Containe

Template



C++ Primer

Jakub Marel

i i ii

landing ove

Typecasts

C--+-:---

.

Template



C++ Primer

Jakub Mare

.. ..

Handing over

Typecasts

Containei

Iterators

Template

we've already seen:



C++ Primer

Jakub Mare

......

Handing over

Typecasts

Container

Itaratara

Iterators

Template:

we've already seen:

implicit typecasting



C++ Primer

Jakub Mare

H. . . P.

Handing over

Typecasts

Containers

.

Iterators

Templates

we've already seen:

implicit typecasting

$$int i = 'b';$$



C++ Primer

Jakub Mare

.. ..

Handing over

Typecasts

Containers

_ .

we've already seen:

implicit typecasting

$$int i = 'b';$$

explicit typecasting

```
C++ Primer
```

Jakub Mare

Revision

rianding ove

Typecasts

Containers

Iterators

Tamalata

we've already seen:

implicit typecasting

```
int i = 'b';
```

explicit typecasting

```
std::cout << (char) i << std::endl;
std::cout << char(i) << std::endl;
```

```
C++ Primer
```

Jakub Mare

Heading area

Handing ove

Typecasts

Containers

Itorators

Templates

```
we've already seen:
```

■ implicit typecasting

```
int i = 'b';
```

explicit typecasting

```
std::cout << (char) i << std::endl;
std::cout << char(i) << std::endl;
works seamlessly only with plain old data (primitives)</pre>
```



C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Container

Iterators

....

4 D > 4 B > 4 E > 4 E > E 990



C++ Primer

Jakub Marel

Revision

Handing ove

Typecasts

Container

Iterators

.....

Template

four new typecasts



C++ Primer

Jakub Marel

Revision

Handing ove

Typecasts

Container

Iterators

Templates

four new typecasts templates



C++ Primer

Jakub Marel

Revision

Handing ove

Typecasts

Iterator

Template

four new typecasts templates works on pointers & references



C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

Containe

Iterators

Template

C



C++ Primer

Jakub Marel

D. data...

Handing ove

Handing ove

Typecasts

Iterators

Templates

dynamic_cast



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

Container

. .

Templates

dynamic_cast

most powerful



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

Container

Iterators

Template

dynamic_cast

most powerful only successful if you're casting into right class



C++ Primer

Typecasts

■ dynamic cast

most powerful only successful if you're casting into right class

```
Animal *a = new Cat("Moonlight");
Cat *c = dynamic_cast<Cat*>(a);
```



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Турссазаз

Contain

Iterators

Tomplate

dynamic_cast

most powerful only successful if you're casting into right class

Animal *a = new Cat("Moonlight"); Cat *c = dynamic_cast<Cat*>(a);

static_cast



C++ Primer

Typecasts

dynamic cast

most powerful only successful if you're casting into right class

Animal *a = new Cat("Moonlight");Cat *c = dynamic_cast<Cat*>(a);

■ static cast

only checks for compatibility



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Container

Iterators

.....

Templates

dynamic_cast

most powerful only successful if you're casting into right class

Animal *a = new Cat("Moonlight"); Cat *c = dynamic_cast<Cat*>(a);

static_cast

only checks for compatibility ie. are the types related to each other?



C++ Primer

Jakub Mare

Danielan

Handing ove

Typecasts

c . .

.

Iterators

Template

dynamic_cast

most powerful only successful if you're casting into right class

Animal *a = new Cat("Moonlight"); Cat *c = dynamic_cast<Cat*>(a);

static_cast

only checks for compatibility ie. are the types related to each other? programmer has to take care of correctness

C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Containe

Iterators

Template

```
dynamic_cast
```

most powerful only successful if you're casting into right class

Animal *a = new Cat("Moonlight"); Cat *c = dynamic_cast<Cat*>(a);

static_cast

only checks for compatibility ie. are the types related to each other? programmer has to take care of correctness

Animal *a = new Cat("Moonlight"); Bat *b = static_cast<Bat*>(a);



C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

Containe

Iterators

Template

C



C++ Primer

Typecasts

■ reinterpret_cast



C++ Primer

Jakub Marel

Revision

Handing over

Typecasts

Containore

Iterators

Templates

■ reinterpret_cast

will copy anything



C++ Primer

Jakub Mare

Revision

Handing over

Typecasts

Container

la a a a a a a a

Iterators

Template:

■ reinterpret_cast

will copy anything usually platform dependent



C++ Primer

Typecasts

■ reinterpret_cast

will copy anything usually platform dependent you can store pointers as integers etc.



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Typecasts

Contain

Iterator

.

Templates

■ reinterpret cast

will copy anything usually platform dependent you can store pointers as integers etc.

```
Animal *a = new Cat("Moonlight");
Car *c = reinterpret_cast<Car*>(a);
```



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

_ .

Iterators

Templates

■ reinterpret_cast

will copy anything usually platform dependent you can store pointers as integers etc.

```
Animal *a = new Cat("Moonlight");
Car *c = reinterpret_cast<Car*>(a);
```

const_cast



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Typecasis

Containe

Iterator:

Template

reinterpret_cast

will copy anything usually platform dependent you can store pointers as integers etc.

```
Animal *a = new Cat("Moonlight");
Car *c = reinterpret_cast<Car*>(a);
```

const_cast

casts const objects into non-const and vice versa

C++ Primer

Jakub Mare

Revisio

Handina a...

Typecasts

Iterators

Templates

```
reinterpret_cast
```

will copy anything usually platform dependent you can store pointers as integers etc.

```
Animal *a = new Cat("Moonlight");
Car *c = reinterpret_cast<Car*>(a);
```

const_cast

casts const objects into non-const and vice versa

```
int encode(char c) {...}
const char c = 'x';
std::cout << encode(const_cast<char*>(c));
```



Questions?

C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Contain

Itaratara

Template:



C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

Containers

Itanatana

Template

C



C++ Primer

Jakub Marel

Revision

Handina a

Typocasts

Containers

Iterators

Templates

vector dynamic array, adding to the end



C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Containers

14---4---

.....

Template

vector dynamic array, adding to the end deque dynamic array, adding to both sides



C++ Primer

Jakub Marek

Revision

Handing over

Typecasts

Containers

Iterators

.

Templates

vector dynamic array, adding to the end deque dynamic array, adding to both sides list linked list, a lot of adding



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Containers

.

....

Templates

vector dynamic array, adding to the end deque dynamic array, adding to both sides list linked list, a lot of adding set set (only unique members), no order multiset set with duplicities



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Containers

Iterators

Templates

vector dynamic array, adding to the end
deque dynamic array, adding to both sides
list linked list, a lot of adding
set set (only unique members), no order
multiset set with duplicities
map mapping given key to a value, unique keys
multimap duplicate keys allowed



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Containers

Iterators

Templates

vector dynamic array, adding to the end deque dynamic array, adding to both sides list linked list, a lot of adding set set (only unique members), no order multiset set with duplicities map mapping given key to a value, unique keys multimap duplicate keys allowed stack stack LIFO queue queue FIFO



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Containers

Iterators

Templates

vector dynamic array, adding to the end deque dynamic array, adding to both sides list linked list, a lot of adding set set (only unique members), no order multiset set with duplicities map mapping given key to a value, unique keys multimap duplicate keys allowed stack stack LIFO queue queue FIFO priority_queue queue with priority



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Containers

Iterators

```
vector dynamic array, adding to the end
        deque dynamic array, adding to both sides
           list linked list, a lot of adding
           set set (only unique members), no order
      multiset set with duplicities
         map mapping given key to a value, unique keys
    multimap duplicate keys allowed
         stack stack LIFO
        queue queue FIFO
priority_queue queue with priority
        string a string
         rope a string container
```



Questions?

C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

Containers



C++ Primer

Jakub Marel

Revision

Handing ove

Typecasts

- . .

Iterators

Tamalata



C++ Primer

Jakub Marel

Handing ove

Typecasts

_

Iterators

Template

allows you to traverse containers



C++ Primer

Jakub Marel

.. ..

Handing ove

Typecasts

Containers

Iterators

Template

allows you to traverse containers several types of



C++ Primer

Jakub Marel

revision

Handing ove

Typecasts

Containers

Iterators

Templates

allows you to traverse containers several types of each container has it's own



```
C++ Primer
```

Jakub Marel

Revision

Handing ove

Typecasts

c . .

Iterators

```
allows you to traverse containers
several types of
each container has it's own

std :: vector < Animal > a;
std :: vector < Animal > :: iterator it;
for (it = a.begin(); it != a.end() && newpet > *it; it++) {}
a.insert(it, newpet);
```



Questions?

C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

Iterators



C++ Primer

Jakub Marek

1101151011

Handing ove

Typecasts



C++ Primer

Jakub Mare

.. ..

Handing over

Typecasts

Containe

Iterators

Templates

allow you to make classes or functions



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Iterators

Templates

allow you to make classes or functions which work dynamically with different other classes



C++ Primer

Jakub Mare

ivenizion

Handing ove

Typecasts

Containe

Iterators

Templates

allow you to make classes or functions which work dynamically with different other classes or you can pass them any parameters

C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

.

Iterators

Templates

allow you to make classes or functions which work dynamically with different other classes or you can pass them any parameters

```
template < class T> template < class T = int> template < int N> template < class T = char, int N = 10> int func < T> (T param) \{\ldots\} class c \{T \ a, \ b, \ c; \ T[N] \ ar; \};
```



C++ Primer

Jakub Marel

IVENIZIOII

Handing ove

Typecasts

Container

Iterator



C++ Primer

Jakub Mare

IVENIZIOII

Handing ove

Typecasts

Containore

Iterators

Templates

oftentimes template does just fine for most classes



C++ Primer

Jakub Mare

Revision

Handing ove

Typecasts

Iterators

Templates

oftentimes template does just fine for most classes but not for some... $% \label{eq:continuous}$



C++ Primer

Jakub Mare

rtevision

Handing ove

Typecasts

Container

Iterators

Templates

oftentimes template does just fine for most classes but not for some... then, we can specialize



C++ Primer

Jakub Mare

Handing ove

Handing ove

Cantainana

Iterators

```
oftentimes template does just fine for most classes but not for some... then, we can specialize 
template < class T> 
class myobject {T ar [80];};

template <> 
class myobject < bool > {unsigned char ar [10];};
```



Questions?

C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts

_ .



Break!

C++ Primer

Jakub Marek

Revision

Handing ove

Typecasts