

fÃ©vr. 04, 11 17:42	complexe.cc	Page 1/2
<pre> #include&lt;iostream&gt; #include&lt;cstdlib&gt;  #include "complexe.h"  using namespace std;  /// ##### les accesseurs ##### ///  int Complexe::getRe() {     return my_re; }  int Complexe::getIm() {     return my_im; }  void Complexe::setRe(int r) {     my_re = r; }  void Complexe::setIm(int i) {     my_im = i; }  /// ##### les constructeur ##### ///  Complexe::Complexe() {     setRe( 0 );     setIm( 0 ); // initialisation Ã 0 }  Complexe::Complexe(int r, int i) {     setRe( r );     setIm( i ); }  /// ##### les methodes ##### ///  void Complexe::afficher() {     cout &lt;&lt; getRe() &lt;&lt; "+i" &lt;&lt; getIm() &lt;&lt; endl; }  void Complexe::addition( Complexe c, Complexe &amp; r ) {     r.setRe( getRe() + c.getRe() );     r.setIm( getIm() + c.getIm() ); }  void Complexe::soustraire( Complexe c, Complexe &amp; r ) {     r.setRe( getRe() - c.getRe() );     r.setIm( getIm() - c.getIm() ); </pre>		

fÃ©vr. 04, 11 17:42	complexe.cc	Page 2/2
<pre> }  void Complexe::multiplier( Complexe c, Complexe &amp; r ) {     r.setRe( (getRe()*c.getRe()) - (getIm()*c.getIm()) );     r.setIm( (getRe()*c.getIm()) + (getIm()*c.getRe()) ); } /*  void diviser( Complexe c, Complexe r );  */ </pre>		

fÃ©vr. 04, 11 18:03

**Makefile**

Page 1/1

```
cc = g++

# RATIONNEL

sourcerationnel = rationnel.cc main.cc util.cc
objetrationnel = $(sourcerationnel:.cc=.o)

sourcecomplexe = testecomplexe.cc complexe.cc
objetcomplexe = $(sourcecomplexe:.cc=.o)

complexe : $(objetcomplexe)
    $(cc) $^ -o $@

rationnel : $(objetrationnel)
    $(cc) $^ -o $@

rationnel.o : rationnel.cc rationnel.h util.h
    $(cc) -c $<

main.o : main.cc rationnel.h util.h
    $(cc) -c $<

util.o : util.cc util.h
    $(cc) -c $<

# COMPLEXE

testecomplexe.o : testecomplexe.cc complexe.h
    $(cc) -c $<

complexe.o : complexe.cc complexe.h
    $(cc) -c $<

#~~~~~

clean :
    rm -vr *.o
```