Heimadæmi 9 - Forritunarmál (Einstaklingsverkefni)

Magnús Daníel Einarsson

October 2023

```
{;;;
Design document for "complex.mmod"
==================================

Use:  z = complex(x,y);
Pre:  x and y are real numbers.
Post: z is the list where head is real number and tail is imaginary.

Use:  x = real(z);
Pre:  z is a complex number.
Post: x is the real part of z.

Use:  x = imag(z);
Pre:  z is a complex number.
Post: b is the imaginary part of z.

Use:  print = printNum(z);
Pre:  z is a complex number.
Post: print is a string with head and tail z.

Use:  z = x+++y;
Pre:  x and y are complex numbers.
Post: z is the complex number x+y.

Use:  z = x---y;
Pre:  x and y are complex numbers.
Post: z is the complex number x-y.

Use:  z = x***y;
Pre:  x and y are complex numbers.
Post: z is the complex number x*y.

Use:  z = x///y;
Pre:  x and y are complex numbers.
      y is not zero.
Post: z is the complex number x/y.
;;;}

"complex.mmod" =
{;;;
;;; Data invariant:
```

```
;;;   A complex number z = x+yi, where x and y
;;;   are double numbers, is represented as [x, y].
;;;}

!
{{
createComp =
    fun(x, y)
    {
        return [x, y];
    };

real =
   fun(z)
    {
        return head(z);
    };

imag =
    fun(z)
    {
        return head(tail(z));
    };

printImag =
   fun(z)
    {
        val print = real(z) ++ "+" ++ imag(z) ++ "i";
        return print;
    };

+++ =
    fun(a, b)
    {
        return createComp(real(a) + real(b), imag(a) + imag(b));
    };

--- =
    fun(a, b)
    {
        return createComp(real(a)-real(b),imag(a)-imag(b));
    };

*** =
    fun(a, b)
    {
        return createComp(
            real(a)*real(b)-imag(a)*imag(b),
            real(a)*imag(b)+imag(a)*real(b)
        );
    };

/// =
    fun(a, b)
```

```
    {
        val temp = real(b)*real(b)+imag(b)*imag(b);
        return createComp(
            (real(a)*real(b)+imag(a)*imag(b))/temp,
            (imag(a)*real(b)-real(a)*imag(b))/temp
        );
    };
}};

"complexnum.mexe" = main in
{{
main =
    fun()
    {
        val test1 = createComp(13.0, 37.0) +++ createComp(4.0, 20.0);
        writeln("(13+37i)+(4+20i)=" ++ printImag(test1));

        val test2 = createComp(13.0, 37.0) --- createComp(4.0, 20.0);
        writeln("(13+37i)-(4+20i)=" ++ printImag(test2));

        val test3 = createComp(13.0, 37.0) *** createComp(4.0, 20.0);
        writeln("(13+37i)*(4+20i)=" ++ printImag(test3));

        val test4 = createComp(13.0, 37.0) /// createComp(4.0, 20.0);
        writeln("(13+37i)/(4+20i)=" ++ printImag(test4));

        val test5 = createComp(2.0, 20.0) +++ createComp(4.0, 3.0);
        writeln("(2+20i)+(4+3i)=" ++ printImag(test2));

        val test6 = createComp(2.0, 20.0) --- createComp(4.0, 3.0);
        writeln("(2+20i)-(4+3i)=" ++ printImag(test2));

        val test7 = createComp(2.0, 20.0) *** createComp(4.0, 3.0);
        writeln("(2+20i)*(4+3i)=" ++ printImag(test2));

        val test8 = createComp(2.0, 20.0) /// createComp(4.0, 3.0);
        writeln("(2+20i)/(4+3i)=" ++ printImag(test2));
    };
}}
*
"complex.mmod"
*
BASIS
;
```

```
java -jar morpho.jar -c complex.morpho
Reused 1894 out of 2648 operations, 754 operation objects used.
Reuse ratio is 72%
```

```
java -jar morpho.jar complexnum
(13+37i)+(4+20i)=17.0+57.0i
(13+37i)-(4+20i)=9.0+17.0i
(13+37i)*(4+20i)=-688.0+408.0i
(13+37i)/(4+20i)=1.9038461538461537+-0.26923076923076923i
(2+20i)+(4+3i)=9.0+17.0i
(2+20i)-(4+3i)=9.0+17.0i
(2+20i)*(4+3i)=9.0+17.0i
(2+20i)/(4+3i)=9.0+17.0i
```