关于Rust的第一次自学

注：举例仅显示必不可少的内容

1. 运行
   1. cargo run：进行debug
   2. cargo run --release：不进行debug
2. 基本类型
   1. 数值类型
      1. 带符号整数：i8, i16, i32, i64, isize（与电脑架构有关）
      2. 无符号整数：u8, u16, u32, u64, usize（与电脑架构有关）
      3. 浮点数：f32, f64
   2. 字符串：&str
   3. 布尔类型：ture/false
   4. 字符类型：表示单个 Unicode 字符，存储为 4 个字节
   5. 单元类型: 即 ( ) ，其唯一的值也是 ( )
3. 创建变量
   1. 可变变量
      1. let mut x=5;
      2. let (mut a,mut b)=(5,4);
   2. 不可变变量：let x=5;
   3. 未使用变量：let \_x=5;
   4. 定义类型的变量：let x:( i32/f32)=2/2.0;
4. 输出变量
   1. println!(“{ }”,x);（自动换行）
   2. 输出两位小数：println!(“{ :.2}”,x);
5. 变量屏蔽
   1. { }内不影响外部
6. 程序检查
   1. assert\_eq!( , )；
      1. 若“，”两侧的取值不同，则会发出警告：warning
7. 函数
   1. .pow(n)：的n次方
   2. .sqrt( )：开平方
   3. .len( )：求长度
8. 运算符
   1. + - \* / %
   2. &：相同记1
   3. |：把1全堆上
   4. ^：不同记1
   5. !：取反
   6. <<：前 的最高位左移 后 个数位
   7. >>：前 的最高位右移 后 个数位
9. 序列
   1. for i in 1..5 {println!("{}",i);}——1-4
   2. for i in 1..=5 {println!("{}",i);}——1-5
10. 引入库
    1. num库
       1. 在Cargo.tom的[dependencies]下添加num=”0.4.0”
       2. 在代码区主函数外添加use num::complex::Complex;