Rust 程序语言设计 assignment2

姓名:　　学号:　　成绩:

作业要求:

1. 作业分为选择题、解答题、代码题，其中选择题和解答题给出答案即可，代码题需要同学们在编译器中动手实现，并张贴代码、代码注释和控制台输出截图。**不需要在此文档中作答，请你撰写自己的作业报告即可。**
2. 请将作业报告于 6.4（包含 6.4）之前发送到邮箱 1784112183@qq.com, DDL 不会延期。文件格式为 docx 或 pdf。

1. What is the purpose of an enum?
    a. **Enums allow us to associate methods with our types**
    b. **Enums allow us to define a type by enumerating its possible variants**
    c. **All of the above**
2. Which of the following are characteristics of matching?
    a. **It is going to compare a value against a series of patterns and then execute based on the matching pattern**
    b. **The compiler will not allow us to forget a "none" or "catch all" statement**
    c. **Matches are exhaustive meaning that ever possibility must be exhausted for the code to be valid**
    d. **All of the above**
3. What do generics allow us to have?
    a. **Generics allow us to have stand in types for our concrete types, which allows our code to be able to operate on many different types.**
    b. **Create specific code for one type.**
    c. **Generics represent a capability that can be implemented on many different types.**
    d. **I have no idea.**
4. What are traits?
    a. **A placeholder for a concrete type.**
    b. **A capability, something a type can do, and can be shared with other types.**
    c. **A quality that a person has.**
5. Which of the following are characteristics of closures?
    a. **Anonymous functions that we can save inside a variable or pass as an argument to other functions**
    b. **The compiler is able to infer the types of the parameters passed into a closure**
    c. **Closures take advantage of inlining, which is a technique where the compiler will add the closure inside the calling functions block**
    d. **All of the above**

What best describes a pointer?

    a. **A variable**

   b. **Data inside a variable**
   c. **The memory address of some data**
   d. **None of the above**

7. Which symbol dereferences a pointer?
   a. **&**
   b. **.**
   c. *****
   d. **^**

8. Are pointer addresses stored on the heap or the stack?
   a. **Heap**
   b. **Stack**

9. What is the difference between Rc and Arc?
   a. **Rc is allocated on the stack and Arc on the heap**
   b. **Rc is not thread safe and Arc is thread safe**
   c. **Rc copies values and Arc clones values**
   d. **None of the above**

10. What is the difference between concurrency and parallel programming?
   a. **Concurrency is the ability for different parts of a program to execute independently. Parallel programming is where different parts of a program execute at the same time**
   b. **Parallel programming is the ability for different parts of a program to execute independently. Concurrency is where different parts of a program execute at the same time**

11. What are channels?
   a. **Message passing where threads communicate by sending each other messages containing data through two ends, a receiver and transmitter**
   b. **A one way flow of data between threads**
   c. **None of the above**

12. Why are mutexes important?
   a. **It allows data to be manipulated by many threads at the same time.**
   b. **It does not allow any thread to access data**
   c. **Allows only one thread access to some data as long as the thread obtains the lock**

13. Smart pointers 是什么？它和 reference 的区别是什么？并简要阐述 Box<T>、Rc<T>、Ref<T>的区别。

14. Rust 的错误处理机制包括哪些？它们都分别用于什么情况？

15. Macros 是什么？请你说明在 rust 中 macros 和 function 的相同点和不同点，并简要阐述 Rust 和 C 语言中 macros 的区别。

16. 在 Rust 中，迭代器提供了一个灵活和强大的方式来处理集合中的数据。本题要求你使用迭代器对 `Vec<String>` 类型的向量进行更高级的操作，完成以下任务：

a. 创建一个包含多个字符串的向量 `words`。每个字符串代表一个单词。

b. 使用迭代器过滤出所有长度大于 3 的单词。

c. 将这些单词转换为全大写。

d. 计算并打印转换后单词的平均长度。

e. 将这些单词按字母顺序排序，并返回一个新的向量。

要求

- 尽量使用迭代器方法链来实现这一功能。

- 平均长度的计算结果应该是 `f64` 类型。

- 注意处理空向量的情况，避免除以零错误。

17. 实现一个简单的社交网络，其中一个用户可以关注多个用户。每个用户包含其名字以及关注的用户列表。要求：

a. 使用 Box<T> 在堆上分配用户结构。

b. 使用 Rc<T> 来实现用户的多重所有权，以便一个用户可以被多个用户关注。

c. 使用 RefCell<T> 来实现用户列表的内部可变性，以便能够修改关注的用户列表。

具体步骤如下：

定义一个 User 结构体，包含 name（String 类型） 和 friends （RefCell<Vec<Rc<User>>> 类型）；实现一个 User 的方法 follow，用于让一个用户关注另一个用户；实现一个 User 的方法 show_friends，用于打印用户的所有关注者的名字。

要求写出完整的代码，并确保代码能够正确编译和运行。

18. 有五位哲学家围坐在一张圆桌旁，桌上放有五根筷子。每位哲学家只有在同时拿起他左边和右边的筷子时才能进餐。每位哲学家进餐完毕后，需要放下他使用的两根筷子，然后继续思考。为了防止死锁，确保程序能够正确实现哲学家就餐过程，使得所有哲学家能够公平地进餐。

要求：

a. 使用多线程模拟哲学家的行为。

b. 使用互斥锁 (Mutex) 来管理筷子的使用。
c. 实现一个函数 dine，该函数启动五个哲学家线程，并保证不会发生死锁。
d. 每位哲学家在思考和进餐时需要打印相应的状态。