

# Threads, exclusión mutua y locks

Técnicas de Programación Concurrente

# Problema

Un viejo banquero retirado se mudó a vivir al campo lejos de toda la tecnología.

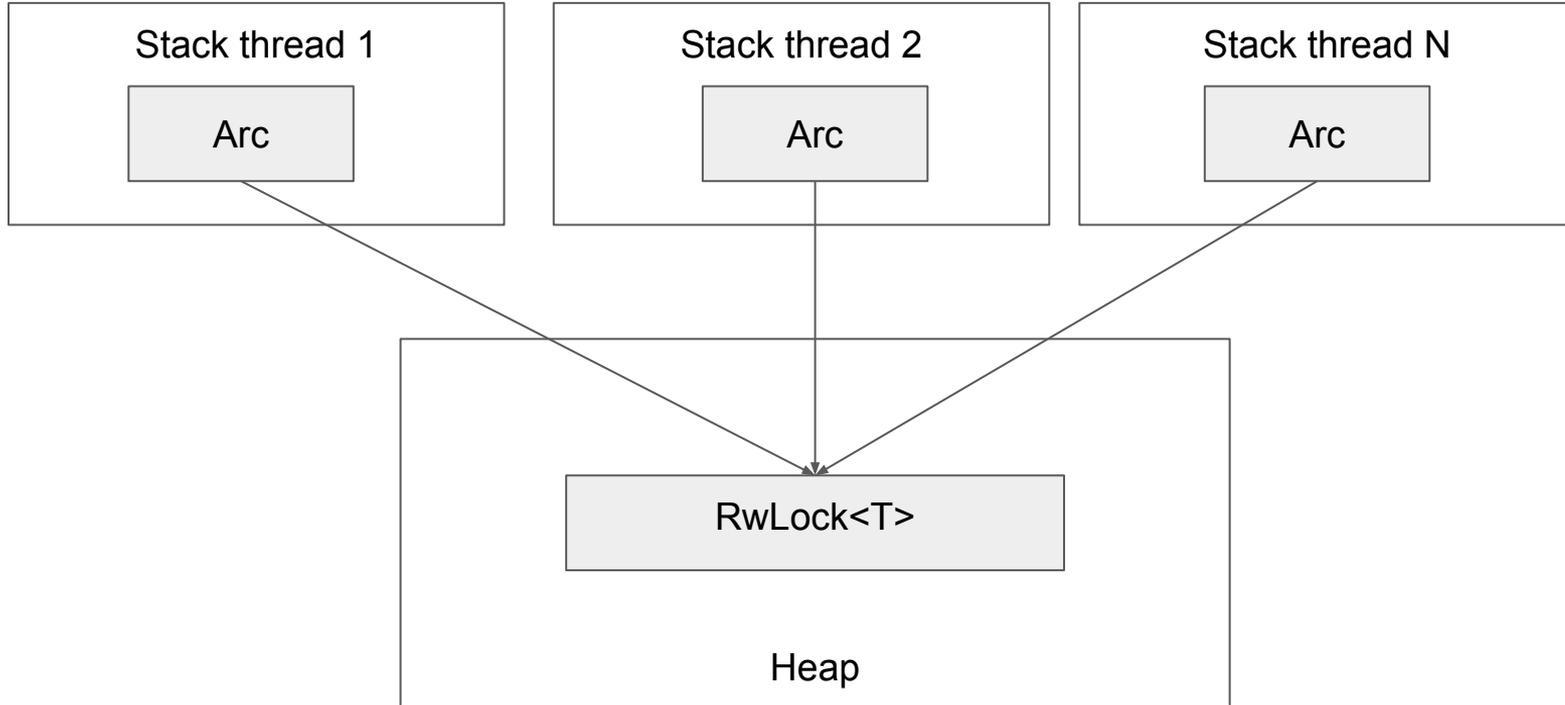
Para vivir, invierte la plata que hizo durante sus años de trabajo mediante los amigos que tiene en diversos fondos de inversión.

Al inicio de cada semana les envía por correo el dinero para invertir a sus amigos; luego espera hasta el final de la semana a que le envíen a su buzón el resultado de esas inversiones.

Modelar la situación planteada en Rust, considerando que:

- A todos los amigos les envía el mismo monto
- Que el dinero resultante lo vuelve a invertir la próxima semana
- Que las inversiones pueden dar una ganancia entre -50% y 50% de lo invertido.

# Funcionamiento de Arc



## Problema 2

Al tiempo el señor fallece.

Los hijos deciden que los inversores sigan trabajando el dinero con algunas condiciones:

- Ellos no se hacen cargo de nada, los inversores solos toman dinero de la cuenta y lo devuelven al final de la semana
- Cada inversor puede reinvertir el capital y hasta 50% de la ganancia propia de la semana anterior, o bien todo el capital en caso de haber perdido.
- Las inversiones deberán ser menos riesgosas, pudiendo dejar de -10% a +10%

# Tarea

Realizar un pequeño *idle game*

- Algunas entidades "extraerán" (generarán aleatoriamente) oro
- Algunas entidades podrán convertir oro en recursos (a gusto)
- Otras entidades podrán convertir combinaciones de recursos en + oro
- Otras entidades podrán solamente consumir recursos
- Periódicamente se reporta por pantalla el nivel de recursos y oro