PdeP - ST - Parcial

Reglas de juego

1. El parcial empieza a las 14:15. No lleguen más tarde porque el horario de finalización se va a mantener y los deja con menos tiempo. Conviene que lo hagan incluso unos minutos antes.
2. La duración del examen es de 3 horas. Lo que no se haya subido antes de las 17:15 (según horario de GitHub) va a ser ignorado.
3. Se va a entregar el enunciado a las 14.15.
4. Se va a publicar la URL con una tarea de GitHub Classroom a través de Discord y escrita en el pizarrón. Ustedes tienen que clonar el repo que se crea al aceptar la tarea, de la misma forma que hicieron con el TP.
5. Lo primero que tienen que hacer en el código es completar su nombre y legajo en el archivo **parcial.hs** con el código del modelo, reemplazando según corresponda en este bloque:  
   *{-  
   Nombre: Apellido, Nombre*

*Legajo: 999999-9*

*-}*

1. Si hay que hacer una salvedad, aclaración o corrección en el enunciado, se ajustará el enunciado a través del pizarrón con un comunicado general, resaltando dónde se cambió.
2. Se va a indicar que hagan commits y pushes cada cierto intervalo de tiempo. El primero a los 20 minutos, luego cada 20 min.
3. No importa el estado intermedio (léase: “funcione o no funcione el código tal cual está”), cuando avisamos que hagan un push, graban los cambios y hacen el *add-commit-push* inmediatamente. Esto toma menos de 1 minuto en total, y pueden empezar a trabajar inmediatamente apenas inician el push y ponen usuario y contraseña (si aplica).
4. Ante cualquier duda con el examen, vamos a implementar una defensa oral del mismo, en fecha a coordinar.
5. Para cada punto del examen, antes de la función principal que implementa ese punto, poner un breve comentario indicándolo. Por ejemplo:  
   *-- Punto 1  
   armarSanguchito ingredientes = … BLAH …*
6. Si no sale un punto, pero la función a implementar es necesaria para resolver otro punto subsiguiente, vale escribir el tipo de la función y una implementación de mentira (que respete ese tipo, pero no haga nada) y luego simplemente usarla. No se va a penalizar la falta de implementación para el punto subsiguiente siempre que la función no resuelta esté bien usada.