



## Programación Concurrente y Distribuida (CC65)

### Trabajo Parcial 2025-2

Secciones: Todas

Profesores: Ms. Carlos Alberto Jara García

### Instrucciones

- Desarrollar uno de los algoritmos según indicación de su profesor.
- El trabajo es por grupo de alumnos máximo 3 integrantes
- Se usará software para detección de plagio.

### Contexto

El trabajo parcial consiste en desarrollar una aplicación que usando GO para recomendar videojuegos de la plataforma Steam utilizando el enfoque de filtrado colaborativo basado en usuarios. La idea central es calcular la similitud entre perfiles de usuarios (según sus reseñas, calificaciones y horas de juego) para sugerir títulos que otros jugadores con preferencias similares han disfrutado.

Dado que el cálculo de similitudes entre usuarios en un dataset masivo puede ser altamente costoso, se propone implementar un algoritmo concurrente que distribuya estas comparaciones en múltiples goroutines, de manera que se optimice el tiempo de ejecución y se mejore la escalabilidad del sistema.

### Descripción de la tarea

El trabajo está dividido en 3 entregables:

- Entregable 1 (PC2), comprende en realizar una revisión bibliográfica de los conceptos teóricos sobre filtrado colaborativo y concurrencia y sus aplicaciones en sistemas de recomendación, usando papers, revistas de investigación y otros medios digitales para elaborar un informe explicativo.
- Entregable 2 (PC2), comprende en realizar una revisión bibliográfica y una prueba de concepto escrita en lenguaje go (secuencial y concurrente) de los siguientes algoritmos recomendados para similitud; analice, realice pruebas de concepto para calcular el tiempo de ejecución (secuencial vs concurrente) y speedup (T-Secuencial / T-Concurrente) y concluya cual es el algoritmo elegido para su trabajo parcial (entregable 3):
  - Cosine Similarity
  - Pearson Correlation
  - Jaccard Index
- Entregable 3 (TP), comprende en implementar el motor de recomendaciones concurrente basado en similitud de usuarios usando el lenguaje go (sin uso de librerías de terceros), tomando como base los entregables 1 y 2. Se desarrollará los siguientes puntos:
  - Diseñe la arquitectura del sistema y flujo concurrente
  - Procese perfiles de usuarios del dataset Steam Reviews 2021 (<https://www.kaggle.com/datasets/najzeko/steam-reviews-2021/data>).

- Calcule en paralelo la similitud entre usuarios, en función al análisis y elección del algoritmo del entregable 2.
- Genere recomendaciones personalizadas de juegos. Cálculo de predicciones.
- Comparar el rendimiento entre las versiones secuencial y concurrente del sistema.
  - Medir métricas de eficiencia: tiempo de cómputo, speedup, scalability, accuracy.
  - Analizar el impacto del número de goroutines en la reducción del tiempo de ejecución.

### **Rúbrica de calificación**

- Entregable 1:
  - (5 puntos) Calidad de la documentación bibliográfica revisada (Papers y artículos científicos de buena reputación)
  - (5 puntos) Análisis y conclusiones de los alumnos.
- Entregable 2:
  - (4 puntos) Implementación de código secuencial y concurrente de los algoritmos. Uso de patrones concurrentes (worker pools, pipelines).
  - (6 puntos) Análisis y conclusiones de pruebas de concepto (evidencias), cálculo de tiempo de ejecución, speedup.
- Entregable 3:
  - (2 puntos) Diseño de la arquitectura del sistema y flujo de concurrencia
  - (2 puntos) Leer el dataset en la aplicación GO
  - (8 puntos) Implementar el algoritmo indicado de manera eficiente (se calificará con 0 puntos si hace uso de librerías de terceros). Uso de patrones concurrentes (worker pools, pipelines).
  - (4 puntos) Evidencia de pruebas, generación de recomendaciones personalizada de juegos, cálculo de predicciones.
  - (4 puntos) Evidencia de análisis y conclusiones de comparar el rendimiento entre las versiones secuencial y concurrente.

### **Documentación**

1.- Presentar un informe entregable PC2 – **Fecha de entrega semana 5:**

- a. Carátula
- b. Resumen del trabajo
- c. Explicación de los algoritmos
- d. Evidencia de pruebas (secuencial y concurrente)
- e. Análisis y conclusiones
- f. Referencias bibliográficas (APA)

2.- Presentar un informe entregable TP – **Fecha de entrega semana 7:**

- a. Carátula
- b. Resumen del trabajo
- c. Diseño de la arquitectura del sistema y flujo de concurrencia
- d. Explicación del algoritmo y el uso de los mecanismos de sincronización utilizado.

- e. Explicación de las pruebas realizadas (tablas, gráficos) comparando secuencial vs concurrente.
- f. Análisis y conclusiones
- g. Referencias bibliográficas (APA)
- h. Subir su código fuente al GitHub (acceso público) y evidenciar los commits en la rama de desarrollo. **Se bajará 5 puntos menos si no se presenta esta evidencia o de identifica que fue editado luego de la fecha de entrega.**
- i. Enlace de video de máximo 5 minutos demostrando cada integrante conocimiento del tema y presentando los resultados del trabajo, discusión crítica sobre limitaciones, escalabilidad y posibles mejoras. (publicarlo en un repositorio en la nube para acceder y visualizar a través de la url). **Se bajará 5 puntos menos si no se presenta el video.**

### **Presentación**

Deberá subir al aula virtual su informe en formato Word. El nombre del archivo deberá tener el formato: CC65-TP-202502-**[Código de alumno]**

Empaquetar en un único archivo zip sus archivos de código fuente y colocar como nombre su Código de alumno (ejemplo 201616054.zip) finalmente subir el archivo al aula virtual.