Título del trabajo – Tema 2: Arrays y Algoritmos de Ordenación

Autor/a: `<tu nombre>` Asignatura: `<asignatura>` Fecha: `<fecha>`

Enlace al vídeo (obligatorio)

- URL YouTube (no listado o público), duración ≥ 10 minutos.
- Debe verse tu cara en miniatura y la ejecución **en vivo** desde VS Code.

1. Introducción

Contexto, objetivos y por qué interesan **estabilidad**, **in-place**, **constantes** y **patrones de entrada**.

2. Metodología

- Hardware/Software (CPU, RAM, SO, Python, VS Code).
- Algoritmos implementados e **instrumentación** (qué se cuenta y cómo).
- Diseño experimental: tamaños, distribuciones, repeticiones, control de ruido.

- Comandos usados: -`python src/bench.py runplot --sizes 500,2000,8000,20000 --reps 3`

3. Trazas manuales

Usa `A = [5, 2, 4, 6, 1, 3, 4]` para:

- Inserción (mostrar estabilidad con duplicados).
- Selección (mostrar inestabilidad potencial).
- Burbuja (+bandera y mejor caso).

4. Resultados experimentales

- Tabla resumen (tiempo, comps, moves) por (algo, dist, n). -**Gráficas** (incluye PNG en `/results/figs`):
  - Tiempo vs n por distribución.
  - Comparaciones y movimientos vs n (random).
  - Quick vs Quick 3-way (dups).
- Comenta anomalías u observaciones destacadas.

5. Discusión

- Constantes y **localidad de caché**: quick vs heap. -**Estabilidad**: cuándo impone usar merge. -**Memoria**: buffer de merge vs in-place (quick/heap). -**Duplicados**: utilidad de 3-way. -**Híbridos**: cortes a inserción; detección de runs.

6. Variantes (elige dos)

Describe, justifica y compara con evidencia:

1. Selection **estable** (sin swaps, con desplazamientos) y su impacto en movimientos.
2. Shell con otra secuencia de gaps (Hibbard/Tokuda/Pratt).
3. Quick **mediana-de-tres** vs **aleatorio**, y profundidad de recursión.
4. Merge **bottom-up** con pequeñas optimizaciones (mini-Timsort).

5. Ordenar **índices** para objetos pesados.

7. Conclusiones

Reglas prácticas de elección según requisitos (estabilidad, memoria, duplicados, tamaño).

8. Reproducibilidad

- Versiones, semilla, comandos, estructura del repo.
- Checklist: asserts, CSV, gráficos, vídeo.

9. Referencias

Cita recursos consultados (libros, apuntes, blogs técnicos, papers, doc. oficial).