

**Universidad Nacional de San Agustín**  
**Escuela Profesional de Ingeniería de Sistemas**  
**Fundamentos de Programación II**  
**Practica de Laboratorio 7:**  
**Combinando Arreglos Estándar y ArrayList**

**I**

---

**OBJETIVOS**

- Crear e inicializar ArrayList
- Realizar búsquedas secuencial y binaria en un ArrayList
- Implementar métodos de ordenamiento en ArrayList
- Combinar arreglos estándar y ArrayList
- Solucionar problemas

**II**

---

**ACTIVIDADES**

1. Cree un Proyecto llamado Laboratorio7
2. Usted deberá crear las dos clases Soldado.java y VideoJuego4.java. Puede reutilizar lo desarrollado en Laboratorios anteriores.
3. Del Soldado nos importa el nombre, puntos de vida, fila y columna (posición en el tablero).
4. El juego se desarrollará en el mismo tablero de los laboratorios anteriores. Para el tablero utilizar la estructura de datos más adecuada.
5. Tendrá 2 Ejércitos (utilizar la estructura de datos más adecuada). Inicializar el tablero con n soldados aleatorios entre 1 y 10 para cada Ejército. Cada soldado tendrá un nombre autogenerado: Soldado0X1, Soldado1X1, etc., un valor de puntos de vida autogenerado aleatoriamente [1..5], la fila y columna también autogenerados aleatoriamente (no puede haber 2 soldados en el mismo cuadrado). Se debe mostrar el tablero con todos los soldados creados y sus puntos de vida (usar caracteres como | \_ y otros, y distinguir los de un ejército de los del otro ejército). Además de los datos del Soldado con mayor vida de cada ejército, el promedio de puntos de vida de todos los soldados creados por ejército, los datos de todos los soldados por ejército en el orden que fueron creados y un ranking de poder de todos los soldados creados por ejército (del que tiene más nivel de vida al que tiene menos) usando 2

diferentes algoritmos de ordenamiento. Finalmente, que muestre qué ejército ganará la batalla (indicar la métrica usada para decidir al ganador de la batalla). Hacer el programa iterativo.

	A	B	C	D	E	F	G	H	I	J
1							1			
2		5		1						
3							1			1
4				1						
5										
6	2					5		2		
7				4						
8	2									
9										
10						1				1