

### Condiciones de entrega:

- Al finalizar deberá subir las respuestas de los ejercicios y un solo archivo `*.py` con las funciones implementadas, en el espacio habilitado en Canvas.
- Asegúrese que el archivo puede abrirse correctamente antes de subirlo a Canvas, ya que de no ser posible verlo, el trabajo se calificará con una nota de cero puntos.
- Se aplicará una penalización del 25 % de la nota por cada día o fracción de entrega tarde. No se aceptará la entrega de trabajos por correo electrónico, ni habrá reposición de esta actividad.
- Se fomenta la colaboración entre estudiantes para el intercambio de ideas o discusiones. Sin embargo, se espera que los estudiantes entreguen sus propias soluciones de los ejercicios. Cualquier sospecha de copia será sancionada con una nota de cero puntos.

Cualquier sospecha de copia o fraude, será sancionada según las **Normas y sanciones académicas**. Pueden consultar el reglamento en el siguiente enlace: [Reglamento de Evaluación de los Aprendizajes](#), artículos 16 a 20.

---

**Instrucciones:** Realice una implementación de un programa de computadora escrito en `Python` que realice lo que se indica.

### Función de dispersión

Implemente un sistema de *hash* (utilizando una función de dispersión  $H(x)$ ) para almacenar números enteros positivos en un arreglo.

**Input:** un arreglo finito  $\mathbf{a} = [a_1, a_2, \dots, a_n]$  de  $n$  enteros positivos y un entero  $m$  tal que  $m \geq n$  (el módulo de la función de dispersión  $H(x)$ ).

**Output:** un arreglo finito  $\mathbf{b}$  con los  $n$  números enteros positivos del arreglo  $\mathbf{a}$  almacenados según la función de dispersión  $H(x)$ .

Nota: Implemente la política para solución de colisiones vista en clase.

### Generador de números pseudoaleatorios

Escriba un programa que genere números pseudoaleatorios usando el método congruencia lineal.

**Input:** cinco números enteros no negativos; el módulo  $m$ , el multiplicador  $a$ , el incremento  $c$ , la semilla  $s$  y la cantidad de números aleatorios a generar  $n$  con  $n \geq 1$ .

**Output:** un arreglo finito  $\mathbf{b}$  de tamaño  $n$  con el listado de números pseudoaleatorios generados.

Observación: No es necesario implementar programación defensiva.

## Ejercicios:

1. Se requiere almacenar los números:

1489, 1237, 1312, 1548, 1209, 853, 519, 992, 339, 535, 883, 1246, 1325, 582, 82, 1517 y 744

utilizando un sistema *hash* con  $m = 17$ .

Escriba su respuesta en la forma:  $[b_0, b_1, \dots, b_{16}]$ .

Nota: Incluya los corchetes y escriba los enteros almacenados separados por coma y sin espacios.

2. Se requiere simular un proceso generando un listado de 20 números pseudoaleatorios, utilizando el método congruencia lineal. Use los siguientes parámetros:

$$m = 23, a = 5, c = 17 \text{ y } s = 2$$

Escriba su respuesta en la forma:  $[r_1, r_2, \dots, r_{20}]$ .

Nota: Incluya los corchetes y escriba los enteros generados separados por coma y sin espacios.

**Importante:** Para poder acceder al cuestionario en Canvas, usted necesita de un código de acceso. El código es el término  $F_{17}$  de la función definida recursivamente como:

$$F_{n+2} = F_n + F_{n+1} \text{ con } F_0 = 1 \text{ y } F_1 = 1, n \geq 0$$