

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA**

ARQUITECTURA DE COMPUTADORAS

**Examen 1
(Segundo semestre 2022)**

Indicaciones generales:

- Duración práctica: 2h.
- Materiales o equipos a utilizar: Con apuntes de clase y acceso a internet.
- Solo está permitida la comunicación con jefes de laboratorio y el profesor del curso. De no seguir esta indicación, se anulará su examen.
- Adjuntar todos sus códigos en un archivo comprimido con el nombre **E1_codigopucp.zip**
- **El archivo en Python debe llamarse programa.py**
- **El archivo en C debe llamarse lib_cs.c**
- **El archivo en ASM debe llamarse contrast_asm.asm**
- **Los incisos a, b y c se califican en el examen.**
- **Los incisos d, e y f deben ser reportados en un PDF con el nombre E1_codigopucp.PDF y debe estar dentro de su archivo en zip.**
- **Cualquier falta a estas indicaciones tendrá un descuento de hasta 30% de su nota final.**

Puntaje total: 10 puntos

Cuestionario:

Pregunta 2 (10 puntos)

El algoritmo de Contrast-Stretching es una operación clásica de procesamiento de imágenes que genera una nueva matriz recibiendo los siguientes argumentos de entrada:

I_{in} : Valor de entrada

I_{out} : Valor de salida de la matriz

$I_{matriz-max}$: Valor máximo de una matriz

I_{min-cs} : Valor del elemento de interés mínimo

I_{max-cs} : Valor del elemento de interés máximo

Cada elemento de la matriz debe seguir la siguiente fórmula:

$$I_{out} = round \left((I_{in}) * \frac{I_{max-cs} - I_{min-cs}}{I_{matriz-max}} \right) + I_{min-cs}$$

Ejemplo:

Considerando que los argumentos de entrada serán los siguientes:

$$I_{min-cs}: 4$$

$$I_{max-cs}: 10$$

y que el argumento de entrada será un vector de elementos enteros entre 0 y $I_{matriz-max}: 15$, el resultado sería el siguiente:

```
Matriz de entrada
[[14 10 9 7]
 [12 15 13 14]
 [ 4 9 0 3]
 [ 2 3 3 0]]
Matriz de salida
[[10 8 8 7]
 [ 9 10 9 10]
 [ 6 8 4 5]
 [ 5 5 5 4]]
```

Por lo tanto, se le pide crear un programa en Python que haga lo siguiente:

- (1.0 punto) Llamar a una función en Python que devuelva un arreglo con la nueva matriz.
- (1.0 punto) Llamar a una función en C que devuelva un arreglo con la nueva matriz.
- (3.0 puntos) Llamar a una función en ASM que devuelva un arreglo con la nueva matriz.
- (0.5 puntos) Codificar un archivo de bash script que ejecute todos sus programas e incluya sus reglas de compilación.
- (2.0 puntos) Realizar el análisis temporal de tiempos de ejecución para un tamaño de matriz $N = \{4, 16, 64, 256, 1024\}$. Reportar una gráfica con los tiempos de ejecución para cada N . Considerar que para cada ejecución de los algoritmos se debe tener un número de iteraciones de 15.
- (1.0 puntos) ¿Cuál codificación ha resultado con mejor rendimiento? Justificar su respuesta
- (1.5 puntos) Explicar cómo impacta la memoria cache en sus resultados. Basar su respuesta en su programación y tamaño de memoria caché.

Consideraciones:

- Matriz es de números naturales.
- Puede llamar al JP al terminar el inciso a), luego para el b) y después para el c). Para el inciso e) puede generar matrices con números aleatorios.

Profesor del curso: Stefano Romero

San Miguel, 10 de octubre de 2022