

# CI0120 Arquitectura de Computadoras ECCI/UCR

## Tarea Corta Optativa

Jorge Chavarría Herrera

B82073

Sergio Martínez Calvo

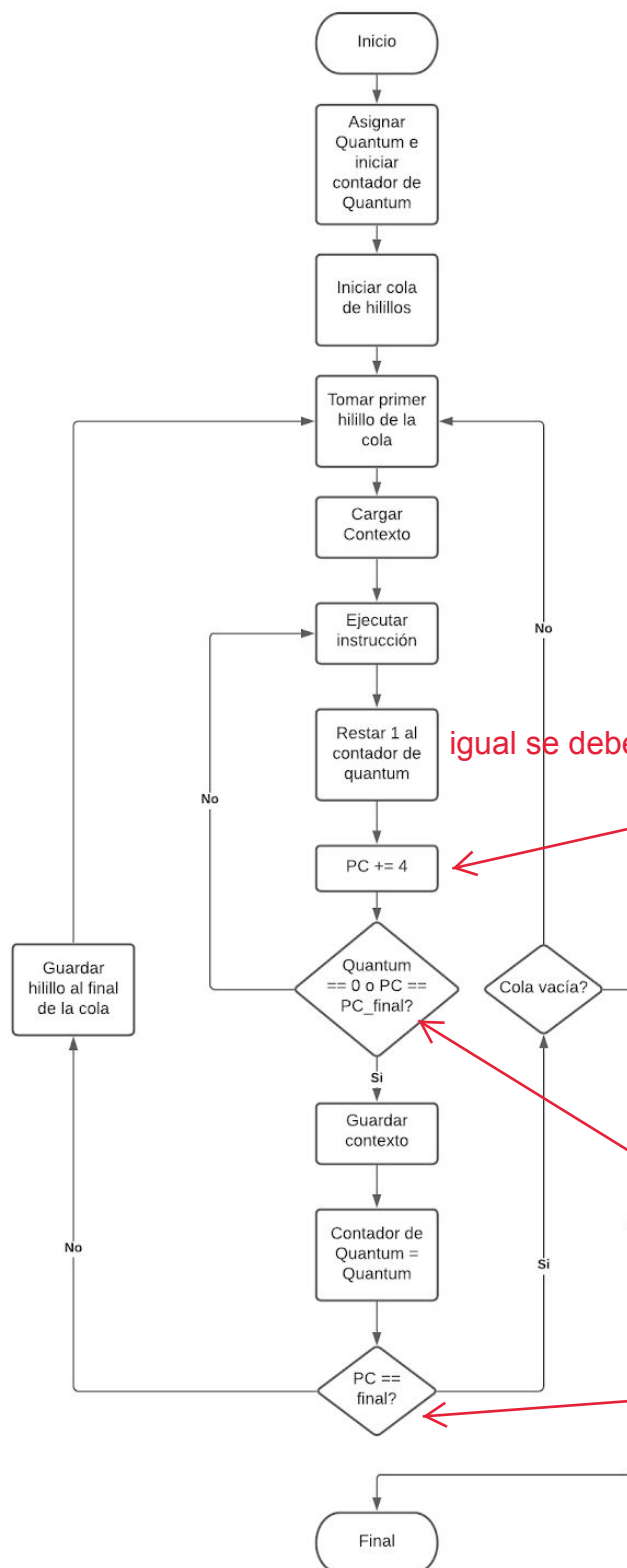
B84621

Rodrigo Vílchez Ulloa

B78292

NOTA: 100 Lo pensaron bien. Hay algunos errores, revisar comentarios.

1. (30 pts) Describir a grandes rasgos con un diagrama la lógica para el **manejo del quantum durante la ejecución de un hilillo**, así como la manera de detectar si se acabó.



igual se debe avanzar ciclo de reloj

Se debe sumar 4 al PC apenas se lee la instrucción, si se hace como acá, se tiene el problema de que los branches y jumps no sirven, porque usan el valor del PC una vez que ya apunta a la siguiente instrucción.

¿PC-final?, no podemos saber el final de un hilillo de antemano. Solo cuando ingresa nuestra instrucción inventada = FIN

2. ( 70 pts) Presentar la lógica (diagrama o pseudocódigo) para **cuando se le acaba el quantum a un hilillo**. Esta lógica debe incluir también

- el caso en el que finalice el hilillo en el mismo ciclo cuando se le acaba su quantum.
- la forma de determinar cuál hilillo sigue, su búsqueda, y la copia de su contexto al procesador para que continúe, o inicie su ejecución.

```
quantum
contador_Quantum = quantum
Memoria_Contexto
cola_hilillos
```

Mientras cola\_hilillos.vacia() == falso hacer:

```
hilillo = cola_hilillos.dequeue()
contexto = Memoria_Contexto.cargarRegistros(hilillo.id)
Por contador en contexto:
    Procesador.registros[contador] = contexto[contador]    ¿?
procesador.ejecutar_instruccion(Procesador.registros[0])
contador_Quantum--
Procesador.registros[0] += 4    problema ya indicado arriba
Si quantum == 0 o hilillo.pc == hilillo.pc_final:    no
```

```
    Memoria_Contexto.guardarContexto(hilillo.id, array_registros_hilillos)    y el PC, se debe
    contador_Quantum = quantum    guardar en el
    Si Procesador.registros[0] != hilillo.pc_final:    No    contexto en dónde
    cola_hilillos.enqueue(hilillo)    se encuentra la
    supongo que es una cola de identificadores de hilillos,    siguiente instrucción
    porque realmente los hilillos no se mueven de memoria.    del hilillo.
```

clase Memoria\_Contexto:

```
    contador_hilillo
    matriz_Contextos
```

```
    guardarContexto(id_hilillo, array_registros_hilillos):
        Por cada columna en matrizContextos[id_hilillo]    bien
            matrizContextos[id_hilillo][columna] = array_registros_hilillos[columna]
        recordar que falta guardar el PC en el contexto
    cargarContexto(id_hili):
```

clase Cola\_Hilillos:

clase Hilillo:

```
        id
        pc_final    no tiene sentido, la última instrucción no es siempre la última en ejecutarse.
        siguienteHilillo
```

```
    cabeza = Null
    ultimo = Null
```

```
    cola_hilillos.enqueue(hilillo):
        ultimo.siguiente = hilillo
        ultimo = hilillo
```

```
    cola_hilillos.dequeue():
        temp = cabeza
```

```
cabeza = cabeza.siguienie  
return temp
```

```
cola_hilillos.vacia():  
    return cabeza==NULL
```