

---

# Proyecto 1

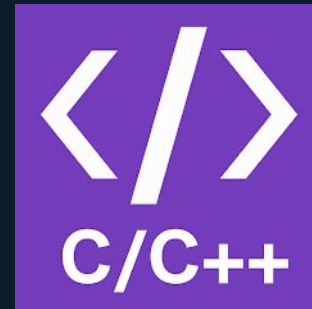
# Procplanner

Juan Esteban Trujillo  
Camilo Alvarez  
Samuel Calderon



# Entorno de Desarrollo

- Sistema operativo: Windows 11 usando WSL (Windows Subsystem for Linux)
- Distribución de Linux en WSL: Ubuntu 24.04.2 LTS
- Compilador utilizado: g++ (Ubuntu 13.3.0-6ubuntu2~24.04) 13.3.0



# Funcionamiento

- Carga procesos desde archivo .txt con PID, registros (AX, BX, CX), quantum y estado inicial.
- Utiliza un planificador Round Robin que ejecuta procesos en turnos fijos, guardando y restaurando su estado en cada cambio de contexto.
- Las instrucciones simuladas (ADD, SUB, MUL, INC, NOP, JMP) se interpretan conforme estén definidas en el .txt de cada proceso
- La ejecución es paso a paso: se actualizan registros, PC y quantum, y se registran los cambios en consola y en un archivo .log.
- El simulador finaliza cuando todos los procesos han completado sus instrucciones o han agotado su quantum o si se bloquea algún proceso, mostrando el estado final de cada proceso.

# Resultados

```
2.txt 3.txt 4.txt 5.txt README.md Informe-Proyecto1-Procplaner.pdf simulacion.log
simulacion.log
1 >> Ejecutando PID 1 durante 8 ciclos o hasta que no hayan mas instrucciones.
2 PID 1 | Ciclo 1 | PC=1 | Quantum=7 | Inst: ADD AX BX
3 PID 1 | Ciclo 2 | PC=2 | Quantum=6 | Inst: SUB BX 2
4 PID 1 | Ciclo 3 | PC=3 | Quantum=5 | Inst: INC CX
5 PID 1 | Ciclo 4 | PC=4 | Quantum=4 | Inst: NOP
6 Proceso 1 ha terminado todas sus instrucciones.
7 [Cambio de contexto] Guardando estado de Proceso 1: PC=4, AX=5, BX=1
8 [Cambio de contexto] Cargando estado de Proceso 2: PC=0, AX=5, BX=2
9 >> Ejecutando PID 2 durante 5 ciclos o hasta que no hayan mas instrucciones.
10 PID 2 | Ciclo 1 | PC=1 | Quantum=4 | Inst: NOP
11 PID 2 | Ciclo 2 | PC=2 | Quantum=3 | Inst: MUL AX 3
12 PID 2 | Ciclo 3 | PC=3 | Quantum=2 | Inst: ADD BX CX
13 PID 2 | Ciclo 4 | PC=2 | Quantum=1 | Inst: JMP 2
14 PID 2 | Ciclo 5 | PC=3 | Quantum=0 | Inst: ADD BX CX
15 Proceso 2 ha agotado su quantum.
16 [Cambio de contexto] Guardando estado de Proceso 2: PC=3, AX=15, BX=10
17 [Cambio de contexto] Cargando estado de Proceso 3: PC=0, AX=8, BX=1
18 >> Ejecutando PID 3 durante 6 ciclos o hasta que no hayan mas instrucciones.
19 PID 3 | Ciclo 1 | PC=1 | Quantum=5 | Inst: ADD CX 1
20 PID 3 | Ciclo 2 | PC=2 | Quantum=4 | Inst: SUB AX BX
21 PID 3 | Ciclo 3 | PC=3 | Quantum=3 | Inst: INC BX
22 PID 3 | Ciclo 4 | PC=4 | Quantum=2 | Inst: NOP
23 Proceso 3 ha terminado todas sus instrucciones.
24 [Cambio de contexto] Guardando estado de Proceso 3: PC=4, AX=7, BX=2
25 [Cambio de contexto] Cargando estado de Proceso 4: PC=0, AX=3, BX=7
26 >> Ejecutando PID 4 durante 4 ciclos o hasta que no hayan mas instrucciones.
27 PID 4 | Ciclo 1 | PC=1 | Quantum=3 | Inst: NOP
28 PID 4 | Ciclo 2 | PC=2 | Quantum=2 | Inst: INC AX
29 [Interrupción] Proceso 4 se bloquea antes de agotar el quantum en ciclo 3.
30 Proceso 4 quedó bloqueado (interrumpido).
31 [Cambio de contexto] Guardando estado de Proceso 4: PC=2, AX=4, BX=7
32 [Cambio de contexto] Cargando estado de Proceso 5: PC=0, AX=0, BX=0
33 >> Ejecutando PID 5 durante 3 ciclos o hasta que no hayan mas instrucciones.
34 [Interrupción] Proceso 5 se bloquea antes de agotar el quantum en ciclo 1.
35 Proceso 5 quedó bloqueado (interrumpido).
36
37 === FIN DE LA PLANIFICACIÓN ===
38
39
```

tru12jt@LaptopJuan:~/Projects/OS/procplaner\$ ./bin/procplaner input/procesos.txt

Procesos cargados desde el archivo:

PID	PC	AX	BX	CX	Quantum	Estado
---	---	---	---	---	-----	-----
1	0	2	3	1	8	Listo
2	0	5	2	4	5	Listo
3	0	8	1	6	6	Listo
4	0	3	7	2	4	Listo
5	0	0	0	0	3	Listo

==== INICIO DE PLANIFICACIÓN ROUND ROBIN ====

>> Ejecutando PID 1 durante 8 ciclos o hasta que no hayan mas instrucciones.

Ciclo 1 | PC = 1 | Quantum restante = 7 | Inst: ADD AX BX

PID	PC	AX	BX	CX	Quantum	Estado
---	---	---	---	---	-----	-----
1	1	5	3	1	7	Ejecutando
2	0	5	2	4	5	Listo
3	0	8	1	6	6	Listo
4	0	3	7	2	4	Listo
5	0	0	0	0	3	Listo

Ciclo 2 | PC = 2 | Quantum restante = 6 | Inst: SUB BX 2

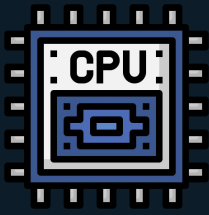
PID	PC	AX	BX	CX	Quantum	Estado
---	---	---	---	---	-----	-----
1	2	5	1	1	6	Ejecutando
2	0	5	2	4	5	Listo
3	0	8	1	6	6	Listo
4	0	3	7	2	4	Listo
5	0	0	0	0	3	Listo

Ciclo 3 | PC = 3 | Quantum restante = 5 | Inst: INC CX

PID	PC	AX	BX	CX	Quantum	Estado
---	---	---	---	---	-----	-----
1	3	5	1	2	5	Ejecutando
2	0	5	2	4	5	Listo
3	0	8	1	6	6	Listo
4	0	3	7	2	4	Listo
5	0	0	0	0	3	Listo

Ciclo 4 | PC = 4 | Quantum restante = 4 | Inst: NOP

PID	PC	AX	BX	CX	Quantum	Estado
---	---	---	---	---	-----	-----
1	4	5	1	2	4	Ejecutando
2	0	5	2	4	5	Listo
3	0	8	1	6	6	Listo
4	0	3	7	2	4	Listo



# Conclusiones



- El simulador muestra claramente cómo el algoritmo Round Robin reparte el tiempo de CPU y gestiona los cambios de contexto.
- Se observa que los procesos pueden ser interrumpidos y bloqueados antes de agotar su quantum, reflejando situaciones reales.
- El registro de ciclos y estados facilita el análisis del comportamiento de los procesos.
- El uso de archivos de texto permite adaptar el simulador a distintos escenarios.
- La eficiencia depende del quantum, las instrucciones y las interrupciones.

