

**Universidad Nacional Autónoma de México**  
**Facultad de Ingeniería**

**TAREA No. 2**

**Comparación de planificadores**

*Profesor(a): ING. GUNNAR EYAL WOLF ISZAEVICH*

*Semestre 2025-1*

Grupo: 6

**Integrantes:**

Gayosso Rosillo Sebastian Emiliano    319191750

Perez Delgado Erandy Estefanya    319147788

Fecha de entrega: 19/11/2024

Diseñar e implementar un programa que permita comparar diversos algoritmos de planificación de procesos, evaluando su desempeño en distintas cargas de trabajo aleatorias mediante métricas de rendimiento como Turnaround Time (T), Tiempo de Espera (E), y Penalización (P), presentando los resultados de forma visual y tabular, incluyendo la integración de algoritmos avanzados como Feedback Multinivel (FB) y Selfish Round Robin (SRR).

```
FCFS
Proceso t      Inicio Fin      T      E      P
A      3      1      4      3      0      1.0
B      4      2      8      6      2      1.5
C      6      4      14     10     4      1.6666666666666667
D      8      6      22     16     8      2.0
E      6      7      28     21     15     3.5
Promedio:      T=11.2 E=5.8 P=1.9333333333333336

Diagrama:
AAA
--BBBB
----CCCCC
-----DDDDDDD
-----EEEEEE

ESQUEMA:
iAAABBBBCCCCDDDDDDDEEEEE
```

```

RR1
Proceso t      Inicio Fin      T      E      P
A      3      1      5      4      1      1.3333333333333333
B      4      2      12     10     6      2.5
C      6      4      23     19     13     3.1666666666666665
D      8      6      28     22     14     2.75
E      6      7      26     19     13     3.1666666666666665
Promedio:      T=14.8 E=9.4 P=2.5833333333333333

Diagrama:
AA-A
-B-B-B---B
--C-C---C--C--C-C
--D---D--D--D--D--D-DD
---E---E--E--E--E-E

ESQUEMA:
iAABAB(CBDCBDCEDCEDCEDCEDD

```

```

RR4
Proceso t      Inicio Fin      T      E      P
A      3      1      4      3      0      1.0
B      4      2      8      6      2      1.5
C      6      4      22     18     12     3.0
D      8      6      26     20     12     2.5
E      6      7      28     21     15     3.5
Promedio:      T=13.6 E=8.2 P=2.3

Diagrama:
AAA
--BBBB
----CCCC-----CC
-----DDDD-----DDDD
-----EEEE-----EE

ESQUEMA:
iAAABBBCCCCDDDEEECCDDDEE

```

```

SPN
Proceso t      Inicio Fin      T      E      P
A      3      1      4      3      0      1.0
B      4      2      8      6      2      1.5
C      6      4      14     10     4      1.666666666666667
D      8      6      28     22     14     2.75
E      6      7      20     13     7      2.166666666666665
Promedio:      T=10.8 E=5.4 P=1.8166666666666669

Diagrama:
AAA
--BBBB
---CCCCC
-----DDDDDDDD
-----EEEEEE

ESQUEMA:
iAAABBBCCCCCCCCCCCCCCCCDDDDDDDD

```

```

SRR
Proceso t      Inicio Fin      T      E      P
A      9      1      17     16     7      1.7777777777777777
B      3      2      19     17     14     5.666666666666667
C     11      4      33     29     18     2.6363636363636362
D      7      8      34     26     19     3.7142857142857144
E      3      9      22     13     10     4.333333333333333
Promedio:      T=20.2 E=13.6 P=3.6256854256854254

Diagrama:
AA-A-A-A-A-A-A-A
-B-B-----B
---C-C-C-C-----C-C-C-C-C-C-C
-----D-D-D-D-D-D-D
-----E-E-E

ESQUEMA:
iAABABACACACADAEBECECDCDCDCDCD

```

## 1. FCFS

Cómo funciona en el código:

- Los procesos se ejecutan completamente uno a la vez, en el orden en que llegan.
- No se permite la interrupción de un proceso en ejecución.
- Comportamiento en el código: Los procesos se ejecutan de forma continua, mostrando un esquema visual lineal que refleja el orden de llegada y ejecución.

## 2. RR

Cómo funciona en el código:

- Cada proceso recibe un máximo de quantum unidades de tiempo.

- Si no finaliza dentro del quantum, se pausa y se coloca al final de la cola.
- Comportamiento en el código: El esquema visual muestra cómo los procesos alternan entre ellos, repitiendo su nombre en intervalos que corresponden al quantum.

### 3. SPN

Cómo funciona en el código:

- Los procesos son ordenados dinámicamente en la cola según su tiempo restante.
  - El proceso más corto siempre se ejecuta primero.
  - No permite interrupciones una vez que un proceso comienza.
- Comportamiento en el código: El esquema visual muestra cómo los procesos más cortos son ejecutados antes, sin importar su orden de llegada, optimizando la eficiencia.

### 4. SRR

Cómo funciona en el código:

- Los procesos comienzan en el nivel más alto de prioridad.
  - Si un proceso no termina dentro del quantum, se mueve a un nivel inferior de prioridad.
  - Los niveles superiores se procesan antes que los inferiores.
- Comportamiento en el código: El esquema visual muestra cómo los procesos con menor prioridad aparecen más tarde y menos frecuentemente.