Operating System

Blen Daniel Assefa

**OS 2021 Problem Sheet #6**

**Problem 6.1: scheduling strategies**

|  |  |  |
| --- | --- | --- |
| Process | Arrival time | Execution time |
| A | 0 | 9 |
| B | 4 | 8 |
| C | 6 | 2 |
| D | 8 | 5 |
| E | 13 | 4 |
| F | 15 | 1 |

1. FCFS

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process | Arrival time | Execution time | Completion time | Turnaround time (Ct - At) or (Wt + Et) | Waiting time (TAT -Et) | Response time |
| A | 0 | 9 | 9 | 9 | 0 | 0 |
| B | 4 | 8 | 17 | 13 | 5 | 5 |
| C | 6 | 2 | 19 | 13 | 11 | 11 |
| D | 8 | 5 | 24 | 16 | 11 | 11 |
| E | 13 | 4 | 28 | 15 | 11 | 11 |
| F | 15 | 1 | 29 | 14 | 13 | 14 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PA | PB | PC | PD | PE | PF | //////// |
| 9 | 17 | 19 | 24 | 28 | 29 |  |

1. SPTF

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process | Arrival time | Execution time | Completion time | Turnaround time (Ct - At) or (Wt + Et) | Waiting time (TAT -Et) | Response time |
| A | 0 | 9 | 9 | 9 | 0 | 0 |
| B | 4 | 8 | 29 | 25 | 17 | 17 |
| C | 6 | 2 | 11 | 5 | 3 | 3 |
| D | 8 | 5 | 16 | 8 | 3 | 3 |
| E | 13 | 4 | 21 | 8 | 4 | 4 |
| F | 15 | 1 | 17 | 2 | 1 | 1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PA | PC | PD | PF | PE | PB | //////// |
| 9 | 11 | 16 | 17 | 21 | 29 |  |

1. LPTF

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process | Arrival time | Execution time | Completion time | Turnaround time (Ct - At) or (Wt + Et) | Waiting time (TAT -Et) | Response time |
| A | 0 | 9 | 9 | 9 | 0 | 0 |
| B | 4 | 8 | 17 | 13 | 5 | 15 |
| C | 6 | 2 | 28 | 22 | 20 | 20 |
| D | 8 | 5 | 22 | 14 | 9 | 9 |
| E | 13 | 4 | 26 | 13 | 9 | 9 |
| F | 15 | 1 | 29 | 14 | 13 | 13 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PA | PB | PD | PE | PC | PF | //////// |
| 9 | 17 | 22 | 26 | 28 | 29 |  |

1. RR

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Process | Arrival time | Execution time | Completion time | Turnaround time (Ct - At) or (Wt + Et) | Waiting time (TAT -Et) | Response time |
| A | 0 | 9 | 22 | 22 | 13 | 0 |
| B | 4 | 8 | 29 | 25 | 17 | 0 |
| C | 6 | 2 | 11 | 5 | 3 | 0 |
| D | 8 | 5 | 27 | 19 | 14 | 0 |
| E | 13 | 4 | 28 | 15 | 11 | 0 |
| F | 15 | 1 | 11 | 14 | 13 | 0 |

Ready queue:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PA | PA | PA | PA | PB | PA | PB | PC | PA | PB | PC | PD | PA | PB | PD | PE | PA | PB | PD | PE | PF | PA | PB | PD | PE | PB | PD | PE | PB |
| 0 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 |

**Problem 6.2: linking**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| file | symbol | internal | external | Weak symbol | Strong symbol |
| a.c | x |  | \* |  |  |
| a.c | y |  |  | \* |  |
| a.c | z | \* |  |  |  |
| a.c | f | \* |  |  |  |
| a.c | g |  |  |  | \* |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| b.c | x |  |  |  | \* |
| b.c | y | \* |  |  |  |
| b.c | z | \* |  |  |  |
| b.c | f |  |  |  | \* |
| b.c | g |  | \* |  |  |

1. The output is an error message. The linking is not possible because of the following reasons:

b.c: f()

a.c: g()

a.c: f()

1. Name mangling is the process of encoding a function or a variable name into a different name so that linkers can separate common names in the language.

Basically, it assists us into changing or encoding similar names of functions and variables from names in the language or even in other files that we are trying to link.

Programming languages like c++ use name mangling because it’s used to facilitate overloading feature and visibility within different scopes.

The “ extern “c” {} ” linkage specifier is declared so as to prevent the C++ compiler from mangling the name of certain functions or declarations.

For example:

extern "C" {

void special\_function(int){

/\* not mangled \*/

}

};

Source: https://www.ibm.com/docs/en/i/7.2?topic=linkage-name-mangling-c-only