Operating Systems

HOST Dispatcher – Practical Marking Schedule

Total Marks: out of 170 - ….% of Assessment **Due Date**: …….

Enough process parameters must be provided to be able to easily assess the process/job being started. If not enough information is available, no marks can be assigned

0. Building program - Party of source code marks (4)

No executable or excess files `[ /2]`
Names as specified: **`ghostd & makefile`** `[ /2]` `[ /4]`

1. Operation of FCFS Real-Time scheduler (10)

1.1 **`fcfs.txt`** - staggered arrival (standard file) `[ /6]`
1.2 **`fcfs1.txt`** - simultaneous arrival `[ /2]`
1.3 **`fcfs2.txt`** - gap in arrival `[ /2]` `[ /10]`

2. Operation of Feedback scheduler in Round Robin mode (10)

2.1 **`rr.txt`** - staggered arrival (standard file) `[ /6]`
2.2 **`rr1.txt`** - simultaneous arrival `[ /2]`
2.3 **`rr2.txt`** - gap in arrival `[ /2]` `[ /10]`

3. Operation of Feedback scheduler (10)

3.1 **`feedback.txt`** - staggered arrival (standard file) `[ /6]`
3.2 **`feedback1.txt`** - Feedback depth `[ /4]` `[ /10]`

4. Mixed dispatcher test (no resource limitations)

4.1 **`mixed.txt`** - Real Time and Feedback mix `[ /5]`
4.2 **`mixed1.txt`** – mixed feedback priorities `[ /5]` `[ /10]`

5. Memory tests (memory limitations only)

5.1 **`memory.txt`** - algorithm type detection (standard file) `[ /6]`
`[ FF ] [ NF ] [ BF ] [ WF ] [ Buddy ]`
5.2 **`memoryexcept.txt`** - memory exceptions `[ /4]` `[ /10]`

6. Resource tests (resource limitations only)

6.1 **`resource.txt`** – test within feedback dispatcher `[ /6]`
6.2 **`resourcexcept.txt`** - resource exceptions `[ /4]` `[ /10]`

7. Combined Test

7.1 **`combined.txt`** - resource exceptions (standard file) `[ /10]`

`Sub-Total:` `[ /74]`

Operating Systems

HOST Dispatcher – Documentation Marking Schedule

Total Marks: out of 170 - ….% of Assessment **Due Date**: …….

- Readability, suitability & maintainability of source code and **`makefile (26+4)`**
  - Student Name & No. on **source file**s & **`makefile`** **`[ /3]`**
  - **`makefile`** hdr comment and layout **`[ /2]`**
  - Code: extensively commented - file hdrs **`[3]`**, fn hdrs **`[3]`**, thru code **`[3]`** **`[ /9]`**
    - and well structured: naming **`[2]`**, fnorder **`[2]`**, layout **`[2]`**, structs **`[2]`** **`[ /8]`**
    - and non-monolithic main **`[2]`**, >1file **`[1]`**, **`.h`** file **`[1]`** (no code in .h) **`[ /4]`**

    `[ /26]`
- Documentation
  - Description, discussion and justification of choice of memory allocation algorithms
    - description: FF, NF, BF, WF, Buddy **`[10]`**
    - discussion, comparison, justification of choice (circle) **`[8]`**
    - **`FF NF BF WF Buddy`** ..... - match **`[2]`** **`[ /20]`**
  - Description and discussion of the structures used by the dispatcher
    - queues generic/specific **`[1]`**
    - process queues & structure **`[1]`**
    - memory arena & structure (single/double link) **`[2]`**
    - resources **`[1]`** **`[ / 5]`**
  - Description and justification of the program structure and individual modules
    - pseudo code + diagram /flow chart **`[5]`**
    - discussion and justification **`[4]`**
    - individual. modules w. descript / in / out / error **`[6]`** **`[ /15]`**
  - Discussion of dispatching scheme, shortcomings, and possible improvements
    - discussion of RT and Feedback dispatching **`[5]`**
    - resource pre-knowledge vs deadlock **`[5]`**
    - memory segment/paging schemes **`[5]`**
    - comparison with UNIX/NT **`[5]`**
    - testing **`[2]`**
    - General Knowledge (subjective judgment) **`[8]`** **`[ /30]`** **`[ /70]`**

`Sub-Total:` **`[ /96]`**

`Sub-Total from functionality tests:` **`[ /74]`**

`Total:` **`[ /170]`**