

Student Math Notes

Generated on 11/21/2025

මෙහෙයුම් පද්ධති ඉගෙනුම් මෙවලම (OS Learning Tool) - සම්පූර්ණ සිංහල උපදේශන සටහන

මෙම මෙවලම (OS Learning Tool) නිර්මාණය කර ඇත්තේ උසස් පෙළ තොරතුරු තාක්ෂණ (ICT) විෂය නිර්දේශයට අදාළ මෙහෙයුම් පද්ධතිවල (Operating Systems) සංකීර්ණ න්‍යායන් සහ ගණනය කිරීම් ප්‍රායෝගිකව අත්හදා බැලීම මගින් පහසුවෙන් අවබෝධ කර ගැනීමටය.

මෙම සටහන, මෙවලමෙහි ඇති සෑම කොටසක්ම (Tab) භාවිතා කරන ආකාරය, න්‍යාය සහ උදාහරණ සහිතව පැහැදිලි කරයි. මෙය ඔබට උපදේශන PDF එකක් ලෙස භාවිත කළ හැක.

I. මෙවලමෙහි මූලික ක්‍රියාකාරීත්වය සහ ව්‍යුහය

කොටස් පාලනය (Collapsible Sections): මෙවලමෙහි ඇති සෑම කොටසකම (Section) ශීර්ෂය මත ක්ලික් කිරීමෙන් අදාළ කොටස විවෘත (Open) හෝ සැඟවීය (Hide) හැක. මෙය ඔබගේ වැඩබිම් (Workspace) පිළිවෙළට තබා ගැනීමට උපකාරී වේ.

ඉගෙනුම් ගලායාම (The "Lane" Concept): සෑම ප්‍රධාන Tab එකක්ම පියවර තුනක ක්‍රමවේදයක් (Input Config Results) අනුගමනය කරයි.

උදව් බොත්තම:

 **Global Help Guide:** මෙවලමෙහි සමස්ත ව්‍යුහය සහ ගලායාම පැහැදිලි කරයි.

Collapsible Help: කොටස් විවෘත කිරීමේ/සැඟවීමේ ක්‍රියාවලිය විස්තර කරයි.

II. CPU Scheduling (CPU කාලසටහන්කරණය)

මෙම කොටස **Process** ක්‍රියාත්මක වන ආකාරය, ඒවා CPU එක සඳහා බලා සිටින කාලය සහ කාර්යක්ෂමතාව ගණනය කිරීමට භාවිතා වේ.

න්‍යායාත්මක පසුබිම

අරමුණ: Average Waiting Time (සාමාන්‍ය රැඳී සිටීමේ කාලය) සහ Average Turnaround Time (සාමාන්‍ය හැරවුම් කාලය) අවම කිරීම.

ප්‍රධාන ඇල්ගොරිතම: FCFS (පළමුව එන්නත්ව පළමුව සේවය), SJF (කෙටිම කාර්යයට පළමුව සේවය), Priority (ප්‍රමුඛතාවය අනුව), RR (Round Robin).

මෙවලම භාවිතා කරන ආකාරය

පියවර	කොටස	ඇතුළු කරන දත්ත	නිරීක්ෂණය කරන න්‍යාය
1	Process Input Configuration	Arrival Time (පැමිණීමේ වේලාව), Burst Time (ක්‍රියාත්මක වීමට අවශ්‍ය කාලය) සහ Priority (ප්‍රමුඛතාවය) ඇතුළු කරන්න.	Burst Time අඩු Process, SJF වලට ප්‍රමුඛත්වය දෙන ආකාරය බලන්න.
2	Algorithm Configuration	අවශ්‍ය ඇල්ගොරිතම (උදා: FCFS, HRRN) තෝරා Calculate All Algorithms ක්ලික් කරන්න.	

		(ප්‍රමුඛතාවය) ඇතුළත් කරන්න.	
2	Algorithm Configuration	අවශ්‍ය ඇල්ගොරිතම (උදා: FCFS, HRRN) තෝරා Calculate All Algorithms ක්ලික් කරන්න.	
3	Scheduling Results (Gantt Charts)	Gantt Chart එක දෙස බලා Process ක්‍රියාත්මක වූ සැබෑ අනුපිළිවෙල සහ CPU Idle Time (CPU හිස්ව තිබූ කාලය) තහවුරු කර ගන්න.	FCFS වලදී P1 දිගු කාලයක් CPU එක රඳවා ගන්නේදැයි නිරීක්ෂණය කරන්න (Convoy Effect).
4	Scheduling Results (Comparison)	Average Waiting Time අඩුම ඇල්ගොරිතම කුමක්දැයි සංසන්දනය කරන්න.	SJF සහ SRTF සාමාන්‍යයෙන් FCFS වලට වඩා හොඳ ප්‍රතිඵල ලබා දෙන ආකාරය බලන්න.

උදාහරණ ගැටළුව සහ ප්‍රශ්නය

දත්ත: P1(AT: 0, BT: 8), P2(AT: 1, BT: 4), P3(AT: 2, BT: 9).

ප්‍රශ්නය: මෙම දත්ත සඳහා **FCFS** සහ **SJF** ඇල්ගොරිතම දෙකෙහි සාමාන්‍ය රැඳී සිටීමේ කාලය (AWT) කොපමණද? (මෙවලම භාවිතයෙන් ගණනය කරන්න.)

III. Process States (ක්‍රියාවලි අවස්ථා සංක්‍රාමණය)

මෙම කොටස **සත් අවස්ථා ක්‍රියාවලි සංක්‍රාන්ති (Seven State Process Transition)** න්‍යාය අවබෝධ කර ගැනීම සඳහාය.

න්‍යායාත්මක පසුබිම

රාජ්‍යයන් (States): New, Ready, Running, Blocked, Ready/Suspend, Blocked/Suspend, Exit.

මාරු කිරීමේ කාර්යයන් (Transitions): Dispatch, Timeout, Suspend, Activate, Event Wait, Event Occurs.

ක්‍රියාකරුවන්: **Long-Term Scheduler** (New Ready), **Short-Term Scheduler** (Ready Running), **Medium-Term Scheduler** (Suspend/Activate).

මෙවලම භාවිතා කරන ආකාරය

පියවර	කොටස	ක්‍රියාත්මක කරන බොත්තම	නිරීක්ෂණය කරන න්‍යාය
1	Process Control Panel	Admit ක්ලික් කරන්න.	Process එක New සිට Ready තත්ත්වයට මාරු වේ (පද්ධතියට ඇතුළත් කිරීම).
2	Available Transitions	Dispatch ක්ලික් කරන්න.	Ready සිට Running තත්ත්වයට මාරු වීම (CPU එක ලබා දීම).
3	Available Transitions	Event Wait ක්ලික් කරන්න.	Running තත්ත්වයේ සිට Blocked තත්ත්වයට මාරු වීම (I/O මෙහෙයුමක් සඳහා ස්වේච්ඡාවෙන් CPU එක අත්හැරීම).
4	Available Transitions	Suspend ක්ලික් කරන්න.	Ready/Suspend තත්ත්වයට යවනු ලබයි (මතකය නිදහස් කිරීමට Hard Disk එක වෙත යැවීම).
5	State Diagram and History	Transition History පරීක්ෂා කර බලා අනුපිළිවෙල සහ සංක්‍රාන්ති වර්ගය තහවුරු කරන්න.	

Diagram and History	කර ලදා අනුප්පලවෙල සහ සංක්‍රාන්ති වර්ගය තහවුරු කරන්න.
---------------------	--

උදාහරණ ගැටළුව සහ ප්‍රශ්නය

තත්ත්ව දාමය: New Ready Running Blocked Ready/Suspend Ready.

ප්‍රශ්නය: මෙම දාමයේ **Blocked Ready/Suspend** යන සංක්‍රාන්තියට වගකිව යුතු Scheduler වර්ගය කුමක්ද? (Hint: එය Suspend කරන නිසා).

IV. Memory Management (මතක කළමනාකරණය)

මෙම කොටස Paging සහ Segmentation, Fragmentation වැනි සංකල්පවල මතක භාවිතය සහ කාර්යක්ෂමතාව ගණනය කිරීමට යොදා ගනී.

තායායාත්මක පසුබිම (Paging & Fragmentation)

Paging: මතකය සමාන ප්‍රමාණයේ කුඩා කොටස්වලට (Pages) බෙදීමෙන් **External Fragmentation** අවම කරයි.

Fragmentation: මතකයේ ඇති නාස්ති වන ඉඩකඩ. **Internal** (partition තුළ ඉතිරිවන ඉඩ) සහ **External** (partition අතර ඇති විසිරුණු ඉඩ).

මෙවලම භාවිතා කරන ආකාරය

පියවර	කොටස	ඇතුළු කරන දත්ත	නිරීක්ෂණය කරන තායාය
1	Memory System Configuration	Address Bits (ලිපින බිටු), Page Size (පිටු ප්‍රමාණය) සහ Page Table Entry Size (PTE ප්‍රමාණය) ඇතුළු කරන්න.	
2	Calculate Memory Overhead	ක්ලික් කරන්න.	Page Table Size (පිටු වගුවට අවශ්‍ය මුළු මතක ප්‍රමාණය) ගණනය කර බලන්න. PTE Size එක විශාල වූ විට Overhead එක වැඩි වන ආකාරය බලන්න.
3	Fragmentation Analysis & Allocation	Job Sizes සහ Partition Sizes ඇතුළු කර Allocation Algorithm (Best/First Fit) තෝරන්න.	First Fit හි වේගය, නමුත් Best Fit හි ඇති අඩු Total Internal Fragmentation නිරීක්ෂණය කරන්න.
4	Address Translation	Logical Address එකක් ඇතුළු කර Translate ක්ලික් කරන්න.	Logical Address එක Page Number සහ Offset ලෙස වෙන්වන ආකාරය අවබෝධ කර ගන්න.

උදාහරණ ගැටළුව සහ ප්‍රශ්නය

දත්ත: Address Bits = 32. Page Size = 4 KB. PTE Size = 4 Bytes.

ප්‍රශ්නය: මෙම පද්ධතියේ **Page Table** එක ගබඩා කිරීමට අවශ්‍ය වන මුළු මතක ප්‍රමාණය (Page Table Size) **MB** වලින් කොපමණද? (මෙවලම භාවිතයෙන් ගණනය කරන්න.)

V. Disk Scheduling (තැටි කාලසටහන්කරණය)

V. Disk Scheduling (තැටි කාලසටහන්කරණය)

මෙම කොටස තැටි හිස (Disk Head) ගමන් කරන කාර්යක්ෂමතාව ගණනය කිරීමට භාවිතා වේ.

තායායාත්මක පසුබිම

අරමුණ: Total Head Movement (මුළු හිස චලනය) අවම කිරීමෙන් Disk Access Time අඩු කිරීම.

ප්‍රධාන ඇල්ගොරිතම: FCFS (සරලම නමුත් අකාර්යක්ෂම), SSTF (කෙටිම සොයන කාලය ප්‍රමුඛ), SCAN (ඉහළ සිට පහළට ගමන් කිරීම).

මෙවලම භාවිතා කරන ආකාරය

පියවර	කොටස	ඇතුළු කරන දත්ත	නිරීක්ෂණය කරන තායාය
1	Disk Request Input	Request Queue (ඉල්ලීම් පෙරලීම) සහ Current Head Position (වත්මන් හිස පිහිටීම) ඇතුළු කරන්න.	
2	Calculate Disk Scheduling	ක්ලික් කරන්න.	
3	 Disk Simulation Results	Total Head Movement අගය සංසන්දනය කරන්න.	FCFS යනු ඇතුළු කළ අනුපිළිවෙල අනුව යන නිසා, අනෙකුත් ඇල්ගොරිතමවලට වඩා මුළු චලනය වැඩි බව නිරීක්ෂණය කරන්න.
4	Sequence of Cylinders	SCAN ඇල්ගොරිතමයේදී හිස ගමන් කරන අනුපිළිවෙල (උදා: 53 → 65 → 67 → 199 → 183...) නිරීක්ෂණය කරන්න.	SCAN හිස සෑම විටම 0 හෝ Max Cylinder දක්වා ගමන් කර දිශාව වෙනස් කරන ආකාරය අවබෝධ කර ගන්න.

උදාහරණ ගැටළුව සහ ප්‍රශ්නය

දත්ත: Current Head Position = 53. Requests: 98, 183, 37, 122.

ප්‍රශ්නය: මෙම දත්ත සඳහා FCFS සහ SSTF ඇල්ගොරිතමවලින් මුළු හිස චලනය (Total Head Movement) අඩුම වන්නේ කුමන ඇල්ගොරිතමයටද? (මෙවලම භාවිතයෙන් ගණනය කරන්න.)

FCFS CPU Scheduling (පළමුව එන්නත්ට පළමුව සේවය) අවබෝධය

1. තායාය: FCFS යනු කුමක්ද?

FCFS (First-Come, First-Served) යනු මෙහෙයුම් පද්ධතිවල ඇති සරලම (simplest) CPU කාලසටහන්කරණ ඇල්ගොරිතම වේ.

ක්‍රියාකාරීත්වය (Logic): Ready Queue එකට පළමුවෙන් පැමිණෙන Process එකට CPU එක පළමුවෙන් වෙන් කර දෙනු ලැබේ.

ස්වභාවය (Nature): FCFS යනු සෑම විටම Non-Preemptive (නො-පෙරළෙන) ඇල්ගොරිතමයක් වේ.

2. මෙවලම භාවිතයෙන් FCFS විශ්ලේෂණය (Tool Simulation)

ඔබේ "  CPU Scheduling " Tab එකේ FCFS ක්‍රියාත්මක වන ආකාරය පහත පියවර අනුව අත්හදා බලන්න.

2. මෙවලම භාවිතයෙන් FCFS විශ්ලේෂණය (Tool Simulation)

ඔබේ " **CPU Scheduling** " Tab එකේ **FCFS** ක්‍රියාත්මක වන ආකාරය පහත පියවර අනුව අත්හදා බලන්න.

පියවර 1: දත්ත ඇතුළු කිරීම (Input Data)

Process Input Configuration කොටසට ගොස් පහත උදාහරණය අනුව දත්ත ඇතුළත් කරන්න:

Process	Arrival Time (පැමිණීමේ වේලාව)	Burst Time (ක්‍රියාත්මක වීමට අවශ්‍ය කාලය)
P1	0	10
P2	1	4
P3	2	2

පියවර 2: ඇල්ගොරිතම තේරීම (Algorithm Selection)

Algorithm Configuration කොටසට ගොස් **FCFS** පමණක් තෝරාගෙන **Calculate All Algorithms** බොත්තම ක්ලික් කරන්න.

පියවර 3: Gantt Chart විශ්ලේෂණය (Visual Analysis)

Scheduling Results කොටසේ **Gantt Charts** Tab එක විවෘත කරන්න.

Gantt Chart: FCFS
P1 (0-10) → P2 (10-14) → P3 (14-16)

නිගමනය: පැහැදිලි කිරීම: P1 මුලින් පැමිණි නිසා, P1, එහි සම්පූර්ණ **Burst Time** එක (10ms) අවසන් වන තුරු CPU එක රඳවා ගනී (Non-Preemptive ස්වභාවය).

3. FCFS හි ගැටළු අවබෝධය (Convoy Effect)

මෙම මෙවලම FCFS හි ඇති ප්‍රධාන අවාසියක් වන **Convoy Effect (ආරක්ෂක රථ ආචරණය)** පහසුවෙන් පෙන්වා දෙයි.

උදාහරණය:

P1 (දිගු, 100ms) මුලින් පැමිණි විට,

P2 සහ **P3** (කෙටි, 1ms බැගින්) වුවද, P1 අවසන් වන තුරු බලා සිටීමට සිදු වේ.

නිගමනය: කෙටි Processes විශාල කාලයක් බලා සිටීම නිසා **Average Waiting Time** ඉහළ යයි. මෙවලමේ **Comparison** Tab එකේදී, FCFS හි Average Waiting Time එක **SJF** වැනි අනෙකුත් ඇල්ගොරිතමවලට වඩා වැඩි බව ඔබට පැහැදිලිව දැකගත හැකිය.