

લિનક્સ ઓપરેટિંગ સિસ્ટમ (4331602) - ગ્રીજ્મ 2025 સોલ્યુશન

Milav Dabgar

13 મે, 2025

પ્રશ્ન 1(a) [3 ગુણ]

ઓપરેટિંગ સિસ્ટમ વ્યાખ્યાયિત કરો અને OS ની જરૂરિયાત સમજાવો.

જવાબ

જવાબ:

ઓપરેટિંગ સિસ્ટમ એ સિસ્ટમ સોફ્ટવેર છે જે કમ્પ્યુટર હાર્ડવેર અને એપ્લિકેશન સોફ્ટવેર વચ્ચે મધ્યસ્થી તરીકે કામ કરે છે. તે હાર્ડવેર રિસોર્સનું સંચાલન કરે છે અને યુઝર પ્રોગ્રામને સેવાઓ પ્રદાન કરે છે.

ઓપરેટિંગ સિસ્ટમની જરૂરિયાત:

- રિસોર્સ મેનેજમેન્ટ: CPU, મેમરી, સ્ટોરેજ અને I/O ડિવાઇસનું કાર્યક્ષમ સંચાલન
- યુઝર ઇન્ટરફેસ: યુઝર ઇન્ટરફેસ માટે કમાન્ડ-લાઇન અને ગ્રાફિકલ ઇન્ટરફેસ પ્રદાન કરે છે
- પ્રોગ્રામ એક્ઝિક્યુશન: યુઝર પ્રોગ્રામને સુરક્ષિત રીતે લોડ અને એક્ઝિક્યુટ કરે છે

મેમરી ટ્રીક

"RUP - રિસોર્સ, યુઝર, પ્રોગ્રામ મેનેજમેન્ટ"

પ્રશ્ન 1(b) [4 ગુણ]

પ્રક્રિયા નિયંત્રણ બ્લોક (PCB) પર એક ટૂંકી નોંધ લખો.

જવાબ

જવાબ:

પ્રોસેસ કન્ટ્રોલ બ્લોક (PCB) એ ડેટા સ્ટ્રક્ચર છે જે ઓપરેટિંગ સિસ્ટમ દ્વારા દરેક ચાલતી પ્રક્રિયા માટે જાળવવામાં આવે છે.

કોષ્ટક 1. PCB ઘટકો

PCB ઘટક	વર્ણન
પ્રોસેસ ID	પ્રક્રિયા માટે અન્ય ઓળખકર્તા
પ્રોસેસ સ્ટેટ	વર્તમાન સ્થિતિ (તૈયાર, ચાલુ, રાહ જોવી)
પ્રોગ્રામ કાઉન્ટર	એક્ઝિક્યુટ કરવાની આગામી instruction નું સંનામું
CPU રજિસ્ટર્સ	પ્રક્રિયા suspend થાય ત્યારે CPU રજિસ્ટર્સની કિમતો
મેમરી મેનેજમેન્ટ	બેઝ અને લિમિટ રજિસ્ટર્સ, પેજ ટેબલ્સ
I/O સ્ટેટ્સ	ખુલ્લી ફાઇલો અને I/O ડિવાઇસની યાદી

મુખ્ય કાર્યો:

- પ્રક્રિયા ઓળખ: અન્ય પ્રોસેસ ID અને પેરેન્ટ પ્રોસેસ ID સ્ટોર કરે છે
- સ્ટેટ ઇન્ફોર્મેશન: વર્તમાન એક્ઝિક્યુશન સ્ટેટ અને કન્ટેક્ટ જાળવે છે
- રિસોર્સ એલોકેશન: ફાળવેલ રિસોર્સ અને મેમરી ઉપયોગનું ટ્રેકિંગ કરે છે

મેમરી ટ્રીક

“PIS - Process ID, Information, State tracking”

પ્રશ્ન 1(c) [7 ગુણ]

વિવિધ પ્રકારની ઓપરેટિંગ સિસ્ટમોની યાદી બનાવો. બેચ ઓપરેટિંગ સિસ્ટમના કાર્યને ચોંચ ઉદાહરણ સાથે સમજાવો.

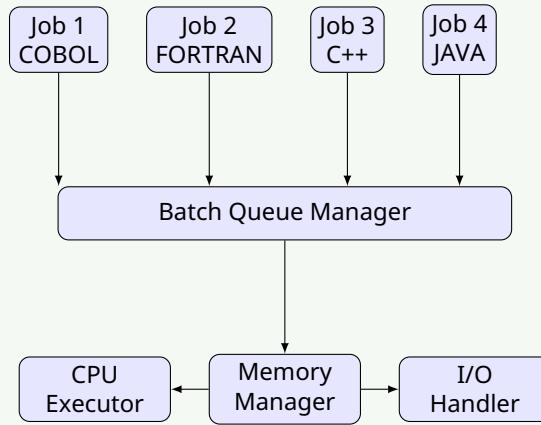
જવાબ

જવાબ:
ઓપરેટિંગ સિસ્ટમના પ્રકારો:

કોષ્ટક 2. ઓપરેટિંગ સિસ્ટમના પ્રકારો

પ્રકાર	વર્ણન
બેચ OS	સમાન જોબ્સને જૂથમાં મૂકીને એકસાથે એક્ઝિક્યુટ કરે છે
ટાઇમ-શેરિંગ OS	બહુવિધ વપરાશકર્તાઓ સિસ્ટમને એકસાથે શેર કરે છે
રીયલ-ટાઇમ OS	નિશ્ચિત રિસ્પોન્સ ટાઇમની ગેરેટી આપે છે
ડિસ્ટ્રિબ્યુટેડ OS	બહુવિધ કનેક્ટેડ કમ્પ્યુટર્સનું સંચાલન કરે છે
નેટવર્ક OS	નેટવર્ક સેવાઓ અને રિસોર્સ શેરિંગ પ્રદાન કરે છે
મોબાઇલ OS	મોબાઇલ ડિવાઇસ માટે ડિઝાઇન કરેલ

બેચ ઓપરેટિંગ સિસ્ટમનું કાર્ય:



આકૃતિ 1. બેચ ઓપરેટિંગ સિસ્ટમ

ઉદાહરણ: બેંક ટ્રાન્ઝેક્શન પ્રોસેસિંગ જ્યાં દિવસભરના બધા ટ્રાન્ઝેક્શન્સ એકત્રિત કરીને રાત્રે કાર્યક્રમતા માટે એકસાથે પ્રોસેસ કરવામાં આવે છે.

મુખ્ય લક્ષણો:

- જોબ ગુપ્પિંગ: કાર્યક્રમતા માટે સમાન જોબ્સ એકસાથે એક્ઝિક્યુટ કરવામાં આવે છે
- કોઈ યુઝર ઇન્ટરેક્શન નહીં: એકવાર સબમિટ કર્યા પછી જોબ્સ યુઝર દખલ વિના ચાલે છે
- ઉચ્ચ થ્રૂપુટ: સિસ્ટમ ઉપરોગને મહત્તમ બનાવે છે

મેમરી ટ્રીક

“JNH - Jobs grouped, No interaction, High throughput”

પ્રશ્ન 1(c) OR [7 ગુણ]

વિવિધ પ્રકારની ઓપરેટિંગ સિસ્ટમોની યાદી બનાવો. રીયલ ટાઇમ ઓપરેટિંગ સિસ્ટમ્સ વિગતવાર સમજાવો.

જવાબ

જવાબ:

ઓપરેટિંગ સિસ્ટમના પ્રકારો: (ઉપરની જેમ સમાન ટેબલ)

રીયલ-ટાઇમ ઓપરેટિંગ સિસ્ટમ (RTOS): રીયલ-ટાઇમ OS એ નિર્દિષ્ટ સમય મર્યાદામાં ગેરેટીડ રિસ્પોન્સ પ્રદાન કરે છે.

RTOS ના પ્રકારો:

કોષ્ટક 3. RTOS ના પ્રકારો

પ્રકાર	ડેડલાઇન	ઉદાહરણ
હાર્ડ રીયલ-ટાઇમ	ડેડલાઇન પૂરી કરવી જ જોઈએ	એર ટ્રાફિક કંટ્રોલ, પેસમેકર
સોફ્ટ રીયલ-ટાઇમ	થોડો વિલંબ સહન કરી શકે	વિડિયો સ્ટ્રીમિંગ, ઓનલાઇન ગેમિંગ
ફર્મ રીયલ-ટાઇમ	કબીકભાર ડેડલાઇન મિસ સ્વીકાર્ય	લાઇવ ઓડિયો પ્રોસેસિંગ

લક્ષણો:

- નિધારિત: બધા ઓપરેશન માટે અનુમાનિત રિસ્પોન્સ ટાઇમ
- પ્રાયોરિટી-આધારિત શેડ્યુલિંગ: ઉચ્ચ પ્રાયોરિટી ટાસ્કને તાત્કાલિક ધ્યાન
- ન્યૂનતમ ઇન્ટરપ્ટ લેટન્સી: ઝડપી કટેકસ્ટ સ્વચિંગ ક્ષમતાઓ
- મેમરી મેનેજમેન્ટ: વિલંબ વિના રીયલ-ટાઇમ મેમરી એલોકેશન

એપ્લિકેશન્સ:

- મેડિકલ ડિવાઇસ, ઓટોમોટિવ સિસ્ટમ્સ, ઇન્ડસ્ટ્રિયલ ઓટોમેશન

મેમરી ટ્રીક

"DPMA - Deterministic, Priority-based, Minimal latency, Applications critical"

પ્રશ્ન 2(a) [3 ગુણ]

પ્રોગ્રામ અને પ્રક્રિયા વચ્ચે તફાવત કરો.

જવાબ

જવાબ:

કોષ્ટક 4. પ્રોગ્રામ vs પ્રક્રિયા

પાસું	પ્રોગ્રામ	પ્રક્રિયા
વ્યાખ્યા	ડિસ્ક પર સંગ્રહિત સ્ટેટિક કોડ	એક્ઝિક્યુશનમાં પ્રોગ્રામ
સ્થિતિ	પેસિવ એન્ટિટી	એક્ટિવ એન્ટિટી
મેમરી	કોઈ મેમરી એલોકેશન નહીં	એલોકેટેડ મેમરી સ્પેસ
જીવનકાળ	ડિલીટ થાય ત્યાં સુધી કાયમી	એક્ઝિક્યુશન દરમિયાન અસ્થાયી
રિસોર્સ	કોઈ રિસોર્સ વપરાશ નહીં	CPU, મેમરી, I/O વપરાશ કરે છે

મુખ્ય તફાવતો:

- સ્ટેટિક vs ડાયનેમિક: પ્રોગ્રામ સ્ટેટિક ફાઇલ છે, પ્રક્રિયા ડાયનેમિક એક્ઝિક્યુશન છે
- રિસોર્સ ઉપયોગ: પ્રક્રિયા સિસ્ટમ રિસોર્સનો વપરાશ કરે છે, પ્રોગ્રામ નહીં
- બૃહુવિધ ઇન્સ્ટન્સ્સ: એક પ્રોગ્રામ બૃહુવિધ પ્રક્રિયાઓ બનાવી શકે છે

મેમરી ટ્રીક

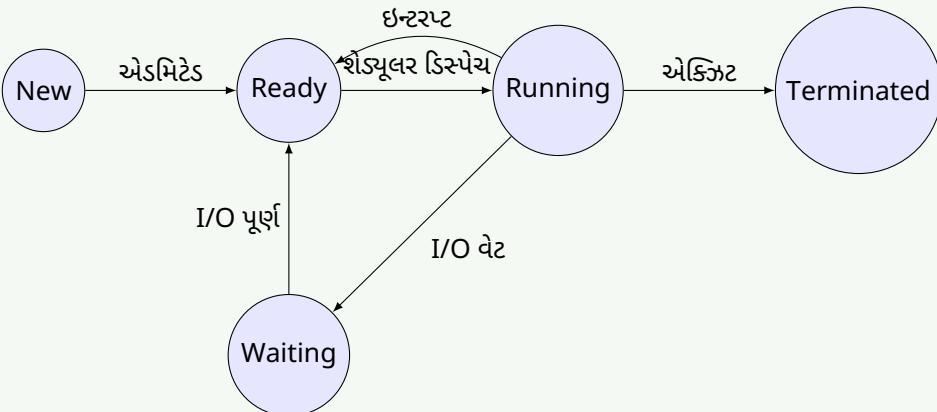
"SDR - Static vs Dynamic, Resource usage, Multiple instances"

પ્રશ્ન 2(b) [4 ગુણ]

પ્રક્રિયા સ્થિતિ રેખાકૃતિની મદદથી પ્રક્રિયાની વિવિધ અવસ્થાઓ સમજાવો.

જવાબ

જવાબ:



આકૃતિ 2. પ્રક્રિયા સ્થિતિ રેખાકૃતિ

પ્રક્રિયા સ્થિતિઓ:

કોષ્ટક 5. પ્રક્રિયા સ્થિતિઓ

સ્થિતિ	વર્ણન
New	પ્રક્રિયા બનાવવામાં આવી રહી છે
Ready	CPU એસાઇનમેન્ટની રાહ જોઈ રહી છે
Running	હાલમાં CPU પર એક્ઝિક્યુશન થઈ રહી છે
Waiting	I/O અથવા ઇવેન્ટ માટે બ્લોક થયેલ
Terminated	પ્રક્રિયાનું એક્ઝિક્યુશન પૂર્ણ થયું

સ્થિતિ પરિવર્તનો:

- Ready થી Running: પ્રોસેસ શેડ્યુલર CPU ફાળવે છે
- Running થી Ready: ટાઇમ સ્લાઇસ સમાપ્ત થાય છે
- Running થી Waiting: પ્રક્રિયા I/O ઓપરેશન માંગે છે
- Waiting થી Ready: I/O ઓપરેશન પૂર્ણ થાય છે

મેમરી ટ્રીક

“NRWRT - New, Ready, Waiting, Running, Terminated states”

પ્રશ્ન 2(c) [7 ગુણ]

રાઉન્ડ રોબિન અલગોરિધમનું વર્ણન કરો અને ગણતરી કરો.

જવાબ

જવાબ:

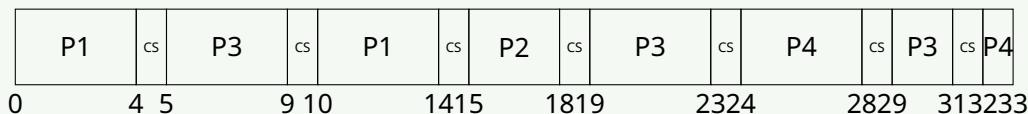
રાઉન્ડ રોબિન અલગોરિધમ: રાઉન્ડ રોબિન એ પ્રીએમિટિવ શેડ્યુલિંગ અલગોરિધમ છે જ્યાં દરેક પ્રક્રિયાને સમાન CPU સમય (કવાન્ટમ) મળે છે.

આપેલ ડેટા:

- કવાન્ટમ ટાઇમ = 4 ms

• કટ્ટેકસ્ટ સ્વિચ = 1 ms

ગેન્ટ ચાર્ટ:



આકૃતિ 3. ગેન્ટ ચાર્ટ (રાઉન્ડ રોબિન)

ગણતરીઓ:

કોષ્ટક 6. ગણતરીઓ

પ્રક્રિયા	કમ્પ્લીશન	ટર્નઅરાઉન્ડ	વેઇટિંગ
P1	14	14 - 0 = 14	14 - 8 = 6
P2	18	18 - 3 = 15	15 - 3 = 12
P3	31	31 - 1 = 30	30 - 10 = 20
P4	33	33 - 4 = 29	29 - 5 = 24

સરેરાશ વેઇટિંગ ટાઈમ = 15.5 ms

સરેરાશ ટર્નઅરાઉન્ડ ટાઈમ = 22 ms

મેમરી ટ્રીક

“FPC - Fair, Preemptive, Context switching overhead”

પ્રશ્ન 2(a) OR [3 ગુણ]

તફાવત કરો: CPU બાઉન્ડ પ્રક્રિયા v/s I/O બાઉન્ડ પ્રક્રિયા।

જવાબ

જવાબ:

કોષ્ટક 7. CPU બાઉન્ડ vs I/O બાઉન્ડ

પાસું	CPU બાઉન્ડ	I/O બાઉન્ડ
પ્રાથમિક પ્રવૃત્તિ	સધન ગણતરીઓ	વારંવાર I/O ઓપરેશન્સ
CPU ઉપયોગ	ઉચ્ચ CPU ઉપયોગ	નીચો CPU ઉપયોગ
બર્સ્ટ ટાઈમ	લાંબા CPU બર્સ્ટ્સ	ટૂંકા CPU બર્સ્ટ્સ
વેઇટિંગ ટાઈમ	ઓછી I/O રાહ	વધુ I/O રાહ
ઉદાહરણો	ગાણિતિક ગણતરીઓ	ફાઇલ ઓપરેશન્સ

મેમરી ટ્રીક

“CIR - CPU intensive, I/O intensive, Resource usage differs”

પ્રશ્ન 2(b) OR [4 ગુણ]

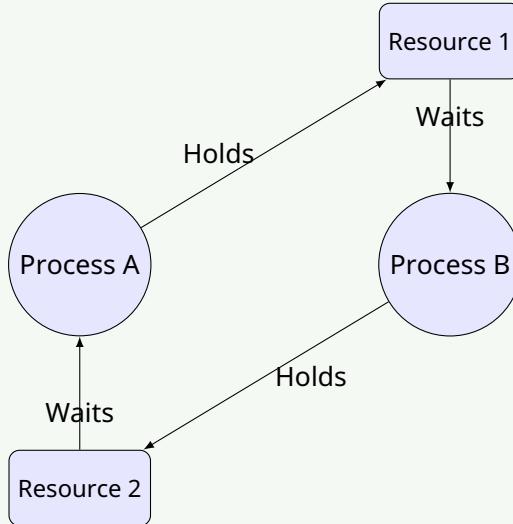
ડેડલોક શું છે? ડેડલોક થવા માટે જરૂરી શરતો સમજાવો.

જવાબ**જવાબ:**

ડેલોક એ એવી પરિસ્થિતિ છે જ્યાં બે અથવા વધુ પ્રક્રિયાઓ કાયમી રૂપે બ્લોક થાય છે.

જરૂરી શરતો (કોફ્નેન કંડિશન્સ):

- મ્યુચ્યુઅલ એક્સક્લુઝન: રિસોર્સ એક્સાથે શેર કરી શકતા નથી
- હોલ્ડ એન્ડ વેઇટ: પ્રક્રિયા રિસોર્સ પક્કીને અન્યની રાહ જુએ છે
- નો પ્રીએમ્પ્લન: રિસોર્સ બળજબરીથી લઈ શકતા નથી
- સર્ક્રૂલર વેઇટ: પ્રક્રિયાઓની વર્તુળાકાર સાંકળ

ઉદાહરણ:

આફ્ટિ 4. ડેલોક

મેમરી ટ્રીક

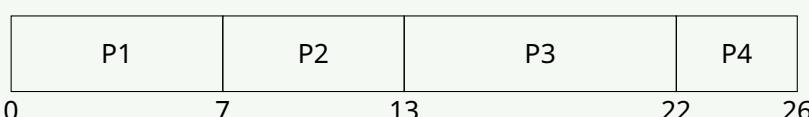
“MHNC - Mutual exclusion, Hold-wait, No preemption, Circular wait”

પ્રશ્ન 2(c) OR [7 ગુણ]

FCFS અલ્ગોરિધમનું વર્ણન કરો અને ગણતરી કરો.

જવાબ**જવાબ:**

FCFS અલ્ગોરિધમ: FCFS એ નોન-પ્રીએમ્પ્લન શેડ્યુલિંગ અલ્ગોરિધમ છે.

ગેન્ટ ચાર્ટ:

આફ્ટિ 5. ગેન્ટ ચાર્ટ (FCFS)

સેરાશ વેઇટિંગ ટાઈમ = 7 ms

સેરાશ ટન્ચારાઉન્ડ ટાઈમ = 13.5 ms

મેમરી ટ્રીક

“SNC - Simple, Non-preemptive, Convoy effect possible”

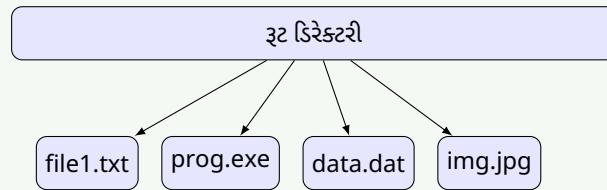
પ્રશ્ન 3(a) [3 ગુણ]

સિંગલ-લેવલ ડિરેક્ટરી માળખું સમજાવો.

જવાબ

જવાબ:

સિંગલ-લેવલ ડિરેક્ટરી સ્ટ્રક્ચર સૌથી સરળ છે જ્યાં બધી ફાઇલો એક જ ડિરેક્ટરીમાં હોય છે.



આકૃતિ 6. સિંગલ-લેવલ ડિરેક્ટરી

મેમરી ટ્રીક

“SUN - Simple, Unique names, No organization”

પ્રશ્ન 3(b) [4 ગુણ]

વિવિધ ફાઇલ લક્ષણો સમજાવો.

જવાબ

જવાબ:

કોષ્ટક 8. ફાઇલ એટ્રિબ્યુટ્સ

એટ્રિબ્યુટ	વર્ણન
નામ	માનવ-વાંચી શકાય તેવું ફાઇલ ઓળખકર્તા
પ્રકાર	ફાઇલ ફોર્મેટ (એક્ઝિટ્યુટિબલ, ટેક્સ્ટ)
કદ	ફાઇલ કદ બાઇટ્સમાં
સ્થાન	સ્ટોરેજ ડિવાઈસ પર સરનામું
પ્રોટેક્શન	એક્સેસ પરમિશન્સ
માલિક	ફાઇલ બનાવનાર યુઝર

મેમરી ટ્રીક

“NTSLPTO - Name, Type, Size, Location, Protection, Time, Owner”

પ્રશ્ન 3(c) [7 ગુણ]

કન્ટ્રીગ્યુઅસ ફાળવણી સમજાવો.

જવાબ

જવાબ:

કન્ટ્રીગ્યુઅસ ફાળવણીમાં, દરેક ફાઇલ ડિસ્ક પર સતત બ્લોક્સનો સેટ વ્યાપે છે.

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

ફાઇલ A

ફાઇલ B

C

આકૃતિ 7. કન્ટીગ્યુઅસ ફાળવણી

મેમરી ટ્રીક

"FMS vs EFC - Fast access, Minimal seek, Simple vs External fragmentation"

પ્રશ્ન 3(a) OR [3 ગુણ]

લિનક્સ ફાઇલ સિસ્ટમના વિવિધ પ્રકારો ટૂંકમાં સમજાવો.

જવાબ**જવાબ:**

કોષ્ટક 9. લિનક્સ ફાઇલ સિસ્ટમ

ફાઇલ સિસ્ટમ	વર્ણન
ext2	જર્નલિંગ નથી
ext3	જર્નલિંગ સાથે
ext4	સુધારેલ પફોર્મન્સ
XFS	ઉર્ચા-પફોર્મન્સ
Btrfs	B-ટ્રી ફાઇલસિસ્ટમ
ZFS	કોપી-એન-રાઇટ

મેમરી ટ્રીક

"EEXBZ - ext2/3/4, XFS, Btrfs, ZFS options"

પ્રશ્ન 3(b) OR [4 ગુણ]

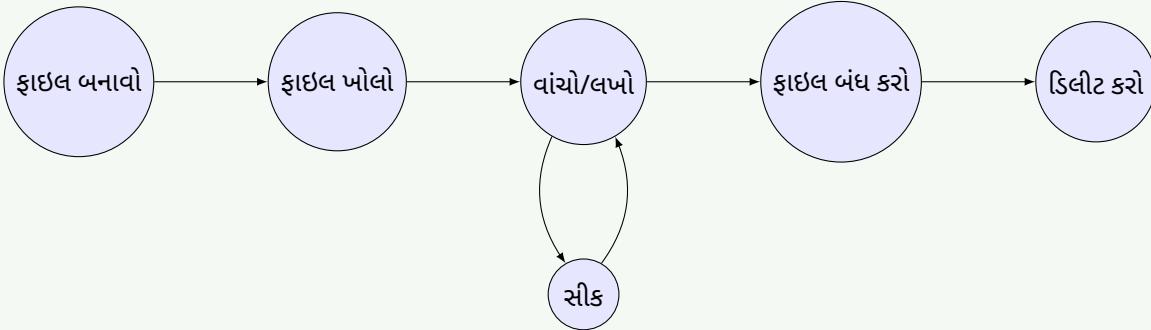
વિવિધ ફાઇલ ઓપરેશન્સ સમજાવો.

જવાબ**જવાબ:**

કોષ્ટક 10. ફાઇલ ઓપરેશન્સ

ઓપરેશન	વર્ણન
બનાવો	નવી ફાઈલ બનાવો
ખોલો	ફાઈલ તૈયાર કરો
વાંચો	ડેટા મેળવો
લખો	ડેટા સ્ટોર કરો
સીક	પોર્જિશન બદલો
બંધ કરો	રિસોર્સ રિલીઝ કરો
ડિલીટ કરો	ફાઈલ દૂર કરો

ફાઈલ ઓપરેશન સિક્વન્સ:



આકૃતિ 8. ફાઈલ ઓપરેશન સિક્વન્સ

મેમરી ટ્રીક

“CORWSCD - Create, Open, Read, Write, Seek, Close, Delete”

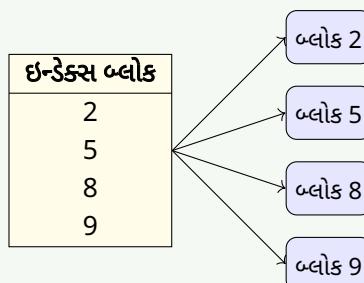
પ્રશ્ન 3(c) OR [7 ગુણ]

ઇન્ડેક્સડ ફાળવણી સમજાવો.

જવાબ

જવાબ:

ઇન્ડેક્સડ ફાળવણીમાં, દરેક ફાઈલ પાસે ડેટા બ્લોક્સના પોઇન્ટર્સ ધરાવતો ઇન્ડેક્સ બ્લોક હોય છે.



આકૃતિ 9. ઇન્ડેક્સડ ફાળવણી

મેમરી ટ્રીક

“NDF vs IMI - No fragmentation, Dynamic size, Fast access”

પ્રશ્ન 4(a) [3 ગુણ]

સિસ્ટમ ધમકીઓ વ્યાખ્યાયિત કરો અને તેના પ્રકારો સમજાવો.

જવાબ

જવાબ:

કોષ્ટક 11. સિસ્ટમ ધમકીઓ

પ્રકાર	વર્ણન
વર્મ્સ	નેટવર્ક પર ફેલાતા પ્રોગ્રામ્સ
વાયરસ	અન્ય પ્રોગ્રામ્સ સાથે જોડાતા કોડ
ટ્રોજન હોર્સ્ટ	છુપાયેલા દુર્ભાવનાપૂર્ણ કાર્યો
DoS	સિસ્ટમ રિસોર્સને ભરાઈ જવાની ડુમલાયો
પોર્ટ સ્કેનિંગ	નેટવર્ક સેવાઓની તપાસ

મેમરી ટ્રીક

“WVTDP - Worms, Viruses, Trojans, DoS, Port scanning”

પ્રશ્ન 4(b) [4 ગુણ]

તફાવત કરો: યુઝર ઓથેન્ટિકેશન V/S યુઝર ઓથોરાઇઝેશન.

જવાબ

જવાબ:

કોષ્ટક 12. ઓથેન્ટિકેશન V/S ઓથોરાઇઝેશન

પાસ્સું	ઓથેન્ટિકેશન	ઓથોરાઇઝેશન
હેતુ	યુઝરની ઓળખ ચકાસવી	યુઝર પરમિશન-સ નક્કી કરવી
ક્યારે	સિસ્ટમ એક્સેસ પહેલાં	ઓથેન્ટિકેશન પછી
પદ્ધતિઓ	પાસવર્ડ્સ, બાયોમેટ્રિક્સ	એક્સેસ કંટ્રોલ લિસ્ટ્સ
પ્રશ્ન	"તમે કોણ છો?"	"તમે શું કરી શકો?"

મેમરી ટ્રીક

“WHO vs WHAT - ઓથેન્ટિકેશન પૂછે છે કોણ, ઓથોરાઇઝેશન નક્કી કરે છે શું”

પ્રશ્ન 4(c) [7 ગુણ]

ઓપરેટિંગ સિસ્ટમ સિક્યુરિટી નીતિઓ અને પ્રક્રિયાઓની ચર્ચા કરો.

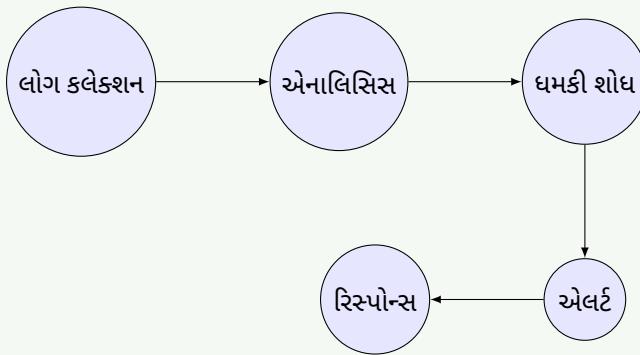
જવાબ

જવાબ:

સિક્યુરિટી નીતિઓ:

- એક્સેસ કંટ્રોલ: કોણ ક્યા રિસોર્સને એક્સેસ કરી શકે
- પાસવર્ડ નીતિ: પાસવર્ડ બનાવતના મુખ્યમો

- ઓડિટ નીતિ: સિસ્ટમ પ્રવૃત્તિઓનું લોગિંગ
સિસ્ટમ મોનિટરિંગ:



આકૃતિ 10. સિક્યુરિટી મોનિટરિંગ

મેમરી ટ્રીક

``AAPUD + UMSIR - Policies + Procedures''

પ્રશ્ન 4(a) OR [3 ગુણ]

પ્રોગ્રામ ધમકીઓ વ્યાખ્યાયિત કરો અને તેના પ્રકારો સમજાવો.

જવાબ

જવાબ:

કોષ્ટક 13. પ્રોગ્રામ ધમકીઓ

પ્રકાર	વર્ણન
મેલવેર	વાયરસ, વર્મ્સ
સ્પાયવેર	યુઝર પ્રવૃત્તિઓનું મોનિટરિંગ
એડવેર	અનિરીદ્ધિત એડવર્ટિઝિંગ
રેન્સમવેર	ડેટા એન્ક્ષિપ્ટ કરે છે
રૂટકિટ્સ	દુર્ભાવનાપૂર્ણ પ્રવૃત્તિઓ છુપાવે છે

મેમરી ટ્રીક

``MSARR - Malware, Spyware, Adware, Ransomware, Rootkits''

પ્રશ્ન 4(b) OR [4 ગુણ]

પ્રોટેક્શન ડોમેનને ચોગ્ય ઉદાહરણ સાથે સમજાવો.

જવાબ

જવાબ:

પ્રોટેક્શન ડોમેન એ ઓપ્ઝેક્ટ્સ અને એક્સેસ રાઇટ્સનો સેટ છે.

ઉદાહરણ:

Domain A

Objects:

- File1 (R,W)
- Printer (W)
- Memory (R,W,X)

Domain B

Objects:

- File2 (R)
- Network (R,W)
- Database (R)

આકૃતિ 11. પ્રોટેક્શન ડોમેન્સ

મેમરી ટ્રીક

``OAS - Objects, Access rights, Subjects define domains''

પ્રશ્ન 4(c) OR [7 ગુણ]

એક્સેસ કંટ્રોલ લિસ્ટ વિગતવાર સમજાવો.

જવાબ**જવાબ:**

એક્સેસ કંટ્રોલ લિસ્ટ (ACL) એ સિક્યુરિટી મેકેનિઝમ છે.

ACL અમલીકરણ:

File: /home/project/report.txt	
User	Permissions
alice	read, write
bob	read
admin	read, write, delete
group:dev	read, write

કોષ્ટક 14. ACL ઉદાહરણ

મેમરી ટ્રીક

``SOA + GDSC''

પ્રશ્ન 5(a) [3 ગુણ]

નીચેના આદેશો સમજાવો: (i) man (ii) cd (iii) ls

જવાબ**જવાબ:**

કોષ્ટક 15. આદેશો

આદેશ	હેતુ	સિન્ટેક્સ
man	મેન્યુઅલ પેજ	man [cmd]
cd	ડિરેક્ટરી બદલો	cd [dir]
ls	લિસ્ટ કર્નેન્ટ	ls [opts] [dir]

મેમરી ટ્રીક

``MCD - Manual pages, Change directory, Directory listing''

પ્રશ્ન 5(b) [4 ગુણ]

ત્રણ સંખ્યાઓ વચ્ચે મહત્વમાન સંખ્યા શોધવા માટે શેલ સ્ક્રિપ્ટ લખો.

જવાબ

Listing 1. મહત્વમાન સંખ્યા

```

1 #!/bin/bash
2 # ત્રણ સંખ્યાઓ વચ્ચે મહત્વમાન સંખ્યા શોધવા માટે સ્ક્રિપ્ટ
3
4 echo "Enter three numbers:"
5 read -p "First number: " num1
6 read -p "Second number: " num2
7 read -p "Third number: " num3
8
9 if [ $num1 -gt $num2 ]; then
10   if [ $num1 -gt $num3 ]; then
11     max=$num1
12   else
13     max=$num3
14   fi
15 else
16   if [ $num2 -gt $num3 ]; then
17     max=$num2
18   else
19     max=$num3
20   fi
21 fi
22
23 echo "Maximum number is: $max"

```

મેમરી ટ્રીક

``ICD - Input, Compare, Display result''

પ્રશ્ન 5(c) [7 ગુણ]

આપેલ 5 અંકની સંખ્યામાં તમામ અંકોનો સરવાળો શોધવા માટે શેલ સ્ક્રિપ્ટ લખો.

જવાબ

Listing 2. અંકોનો સરવાળો

```

1 #!/bin/bash
2 # 5 અંકની સંખ્યાના અંકોનો સરવાળો શોધવા માટે સ્ક્રિપ્ટ
3
4 echo "Enter a 5-digit number:"
5 read number
6

```

```

7 # Validate input
8 if [ ${#number} -ne 5 ] || ! [[ $number =~ ^[0-9]+$ ]]; then
9   echo "Error: Please enter exactly 5 digits"
10  exit 1
11 fi
12
13 sum=0
14 temp=$number
15
16 # Extract and sum each digit
17 while [ $temp -gt 0 ]; do
18   digit=$((temp % 10)) # Get last digit
19   sum=$((sum + digit)) # Add to sum
20   temp=$((temp / 10)) # Remove last digit
21 done
22
23 echo "Number: $number"
24 echo "Sum of digits: $sum"

```

મેમરી ટ્રીક

``VEDS - Validate, Extract, Display, Sum digits''

પ્રશ્ન 5(a) OR [3 ગુણ]

નીચેના આદેશો સમજાવો: (i) date (ii) top (iii) cmp

જવાબ

જવાબ:

કોષ્ટક 16. વધુ આદેશો

Cmd	હેતુ	ઉદાહરણ
date	તારીખ દર્શાવો	date +%F
top	પ્રોસેસ વ્યુ	top
cmp	ફાઇલ સરખામણી	cmp f1 f2

મેમરી ટ્રીક

``DTC - Date/time, Task monitor, Compare files''

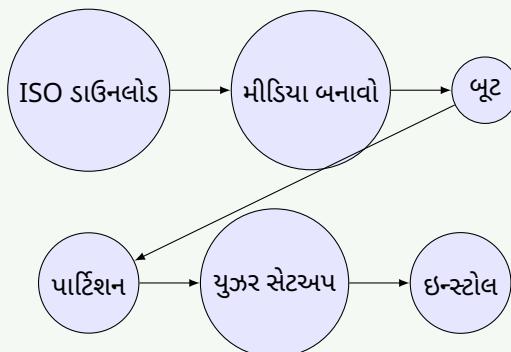
પ્રશ્ન 5(b) OR [4 ગુણ]

લિનક્સના ઇન્સ્ટોલેશન સ્ટેપ્સ સમજાવો.

જવાબ

જવાબ:

ઇન્સ્ટોલેશન પ્રક્રિયા:



આકૃતિ 12. ઇન્સ્ટોલેશન ફ્લો

મુખ્ય સ્ટેપ્સ:

1. પાર્ટિશનિંગ: ડિસ્ક સ્પેસ કન્ફિગર કરો
2. કન્ફિગરેશન: ટાઇમ્પોન, કીબોર્ડ, યુઝર સેટ કરો
3. ઇન્સ્ટોલેશન: ફાઈલો કોપી કરો

મેમરી ટ્રીક

"DCBCPUPI"

પ્રશ્ન 5(c) OR [7 ગુણ]

N સંખ્યાઓનો સરવાળો અને સરેરાશ શોધવા માટે શૈલ સ્ક્રિપ્ટ લખો.

જવાબ

Listing 3. સરવાળો અને સરેરાશ

```

1 #!/bin/bash
2 # Script to find sum and average of N numbers
3
4 echo "How many numbers do you want to enter?"
5 read n
6
7 # Validate input
8 if ! [[ $n =~ ^[0-9]+$ ]] || [ $n -le 0 ]; then
9   echo "Error: Please enter a positive integer"
10  exit 1
11 fi
12
13 sum=0
14 echo "Enter $n numbers:"
15
16 # Read N numbers
17 for ((i=1; i<=n; i++)); do
18   echo -n "Enter number $i: "
19   read number
20   # Simple accumulation
21   sum=$(echo "$sum + $number" | bc -l)
22 done
23
24 # Calculate average
25 average=$(echo "scale=2; $sum / $n" | bc -l)
26
  
```

```
27 echo "Sum: $sum"  
28 echo "Average: $average"
```

મેમરી ટ્રીક

``VLAD - Validate, Loop, Arithmetic, Display''