

ШИНЖЛЭХ УХААН, ТЕХНОЛОГИЙН ИХ СУРГУУЛЬ

МЭДЭЭЛЭЛ, ХОЛБООНЫ ТЕХНОЛОГИЙН СУРГУУЛЬ



БИЕ ДААЛТ I

Үйлдлийн систем (F.CS301)
2023-2024 оны хичээлийн жилийн намар

Бие даалтын ажлын нэр: Процесс ба салбар процесс

Хичээл заасан багш:/О.Бат-Энх/

Бие даалтын ажил гүйцэтгэсэн:

B210910833/Тэмүгэ/

B211900004/Болор-Эрдэнэ/

B211900015/Ганболор/

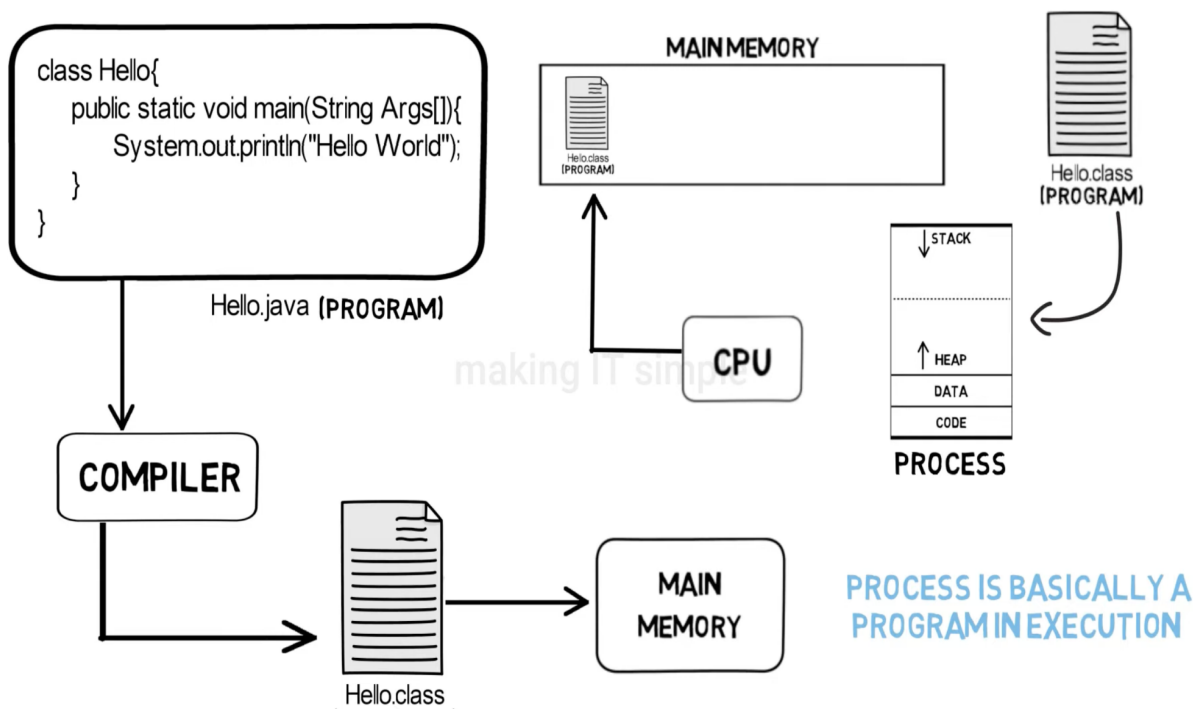
B210910082/Ган-Эрдэнэ/

2023 он

1.Процесс

Процесс гэдэг нь аливаа системд хэрэгжих ёстой ажлын үндсэн нэгжийг төлөөлөхөд тусалдаг ажилын нэгжийг хэлнэ.Процесс нь үндсэндээ програм хангамжийг ажиллуулдаг. Аливаа процессыг гүйцэтгэх нь тодорхой дарааллаар явагдах ёстой.

Бид компьютерийн программуудыг текст файл хэлбэрээр бичдэг бөгөөд ингэснээр тэдгээрийг ажиллуулахад програмд заасан бүх үүргийг гүйцэтгэдэг процесс болж хувирдаг. Програмыг санах ойд оруулахдаа процесс болгон дөрвөн хэсэгт хувааж болно: стек, heap, текст, өгөгдөл.



Процесс нь бүх тооцооллын үндэс суурь болдог гүйцэтгэгдэж байгаа программ юм. Энэ нь компьютерийн кодтой адил биш боловч маш төстэй юм. Идэвхгүй гэж үздэг програмаас ялгаатай нь процесс нь "идэвхтэй" нэгж бүрдэл юм. Техник хангамжийн төлөв, RAM, CPU болон бусад шинж чанарууд нь процесст хамаарах шинж чанарууд юм.

Ямар нэг программ бичих гэж байгаа бол эхлээд дээд түвшиний хэлээ сонгоно ямар ч хэл байсан болно жишээ нь java, c++, python гэх мэт. Бичсэн программ компьютерт ойлгомжтой болгохын тулд компайл хийж binary буюу хоёртын код болгох хөрвүүлэх ёстой бөгөөд энэхүү компайл хийсэн кодыг cpu ажилуулах юм бол энэ нь энгийнээр процесс болно.

2.Процесс загварууд

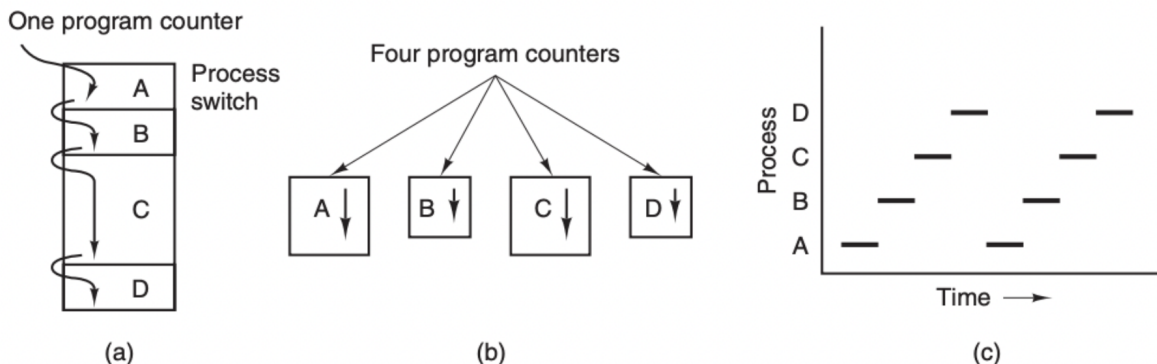


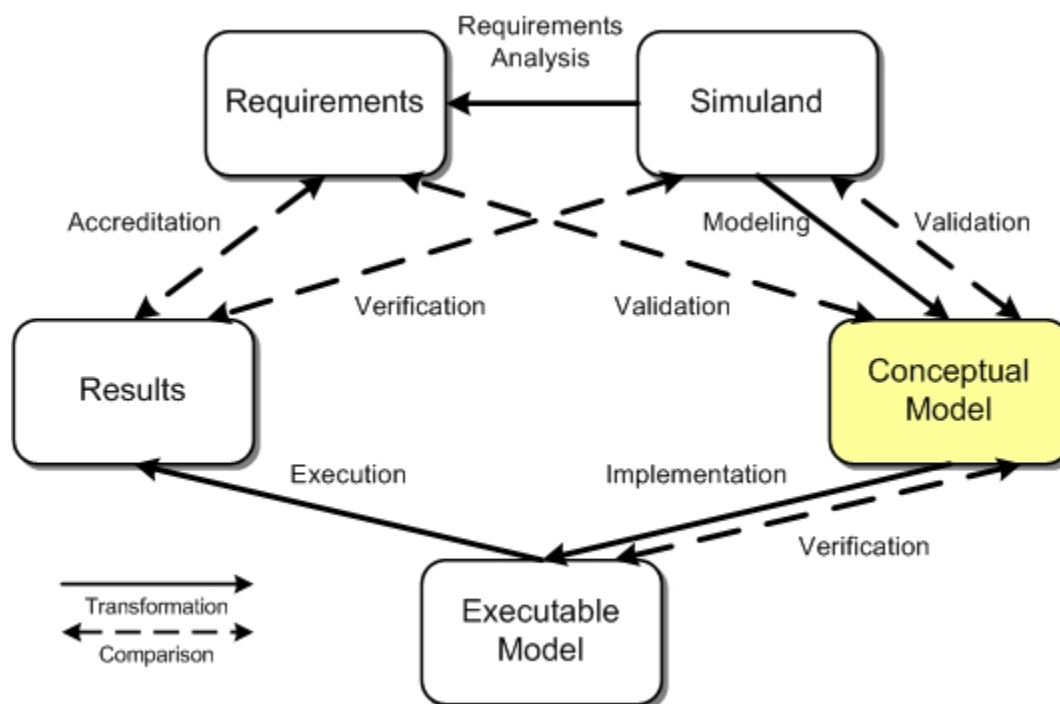
Figure 2-1. (a) Multiprogramming four programs. (b) Conceptual model of four independent, sequential processes. (c) Only one program is active at once.

(a) Multiprogramming four programs

Бүх программыг нэг процессорт нэгэн зэрэг ажиллана. Мультипрограмчлал нь системд үндсэн санах ойд ачаалагдсан нэг буюу хэд хэдэн программууд ажиллахад бэлэн байдаг. Нэг удаад зөвхөн нэг программ нь CPU-г өөрийн зааварчилгааг (өөрөөр хэлбэл систем дээр ажиллаж байгаа хамгийн ихдээ нэг процесс ажиллаж байгаа) авах боломжтой байхад бусад нь ээлжээ хүлээж байна.

Мультипрограмчлалын гол санаа нь программчлалын ашиглалтыг нэмэгдүүлэх явдал юм. CPU-ийн цаг. Үнэн хэрэгтээ, одоо ажиллаж байгаа процесс нь I/O даалгаврыг гүйцэтгэж байна гэж бодъё (тодорхойлолтын дагуу үүнийг гүйцэтгэхэд CPU шаардлагагүй). Дараа нь үйлдлийн систем нь энэ үйл явцыг тасалдуулж, гүйцэтгэхэд бэлэн байгаа бусад үндсэн санах ойн программуудын аль нэгэнд удирдлагыг өгч болно (жишээ нь процессийн контекст шилжих). Ийм байдлаар систем оролт/гаралтын ажлыг дуусгахыг хүлээж CPU-ийн цагийг дэмий үрэхгүй бөгөөд ажиллаж байгаа процесс нь CPU-г сайн дураар гаргах хүртэл эсвэл оролт/гаралтын үйлдлийг блокдох хүртэл үргэлжлүүлэн ажиллана. Тиймээс олон програмчлалын эцсийн зорилго нь гүйцэтгэхэд бэлэн процессууд байгаа үед CPU-г завгүй байлгах явдал юм. Ийм системийг зөв ажиллуулахын тулд үйлдлийн систем нь үндсэн санах ойн тусдаа хэсэгт олон програм ачаалах чадвартай байх ёстойг анхаарна уу. нэг процессыг нөгөө процессоор өөрчлөхөөс зайлсхийхийн тулд шаардлагатай хамгаалалтыг хангана. Санах ойд олон програм байх үед шийдвэрлэх шаардлагатай бусад асуудал бол програмууд үндсэн санах ой руу орох эсвэл гарах үед хуваагдах явдал юм. Мөн шийдвэрлэх шаардлагатай өөр нэг асуудал бол том хэмжээний программууд санах ойд нэг дор багтахгүй байж болох бөгөөд үүнийг хуудаслах болон виртуал санах ой ашиглан шийдэж болно.

(b) Conceptual model of four



J. Sokolowski, C. Banks, Modeling and Simulation Fundamentals: Theoretical Underpinnings and Practical Domains, Wiley, 2010, pp 333

Процессууд тус бүр өөрийн гэсэн хяналтын урсгалтай ба энэ нь бие даасан дараалласан процесстой загвар юм.

Дөрвөн бие даасан дараалсан үйл явцын концепцийн загварыг үйл явц бүр нь бусдаас хамааралгүйгээр ажилладаг боловч ерөнхий зорилго эсвэл үр дүнд хувь нэмрээ оруулах харилцан уялдаатай хэд хэдэн алхам эсвэл үе шат болгон дүрсэлж болно. Эдгээр үйл явц нь тодорхой дарааллаар ар араасаа явагддаг. Ийм системийн ерөнхий үзэл баримтлалын загвар энд байна.

Процесс 1: Энэ нь загвар дахь анхны дараалсан процесс юм. Энэ нь өгөгдөл цуглуулах, оруулах эсвэл эхний үйл ажиллагааг хамарч болно. Энэ процессын гаралт нь дараагийнх нь оролт болно.

Процесс 2: Процесс 1-ийн гаралт нь Процесс 2-д ордог. 2-р процесс нь 1-р процессоос хамааралгүй ажиллаж, өөрийн багц даалгавруудыг гүйцэтгэдэг. Үүнд дүн шинжилгээ хийх, хувиргах эсвэл цаашдын өгөгдөл боловсруулах зэрэг орно.

Процесс 3: Процесс 2-ыг дуусгасны дараа өгөгдөл буюу гаралтыг Процесс 3 руу шилжүүлнэ. Өмнөх процессуудын нэгэн адил бие даан ажиллаж, өөрийн гэсэн тодорхой үйл ажиллагаа эсвэл функцийг гүйцэтгэдэг.

Процесс 4: Дарааллын эцсийн алхам нь Процесс 4. Процесс 3-ын гаралтыг авч бие даасан даалгавраа гүйцэтгэдэг. Энэ үе шатанд бүхэл бүтэн системийн эцсийн зорилго эсвэл үр дүнд хүрч болно.

Энэхүү концепцийн загварын гол шинж чанарууд нь:

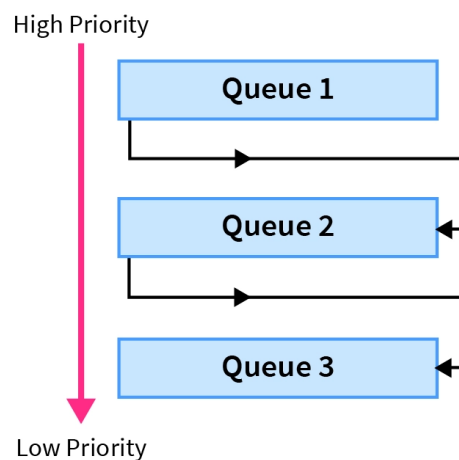
Хараат бус байдал: Үйл явц бүр нь бие даасан бөгөөд дараалсан дарааллаар харилцан уялдаатай боловч бусдын шууд нөлөөгүйгээр ажилладаг.

Дараалсан: Процессууд тодорхой дарааллаар ар араасаа явагдах ба нэг процессын гаралт нь дараагийн процессын оролт болдог.

Харилцан холболт: Процессууд нь өгөгдөл эсвэл мэдээллийн урсгалаар хоорондоо холбогддог боловч бие биенийхээ үйл ажиллагаанд шууд нөлөөлдөггүй.

Энэхүү үзэл баримтлалын загварыг ихэвчлэн үйлдвэрлэл, өгөгдөл боловсруулах, програм хангамж боловсруулах болон бусад олон салбарт ашигладаг бөгөөд тодорхой зорилго эсвэл үр дүнд хүрэхийн тулд тодорхой боловч харилцан уялдаатай хэд хэдэн алхмуудыг хийх шаардлагатай байдаг.

(c) Only one program is active at once:



Тухайн агшинд зөвхө нэг процесс ажиллах бөгөөд хэсэг хугацааны дараа дараагийн процесс ажиллах зарчимаар явагдана.

Үйлдлийн систем нь процессор руу нэвтрэх дараалал, эдгээр процесс даалгавар тус бүрийг боловсруулах хугацааг шийдэх гол үүрэгтэй. Үйлдлийн системийн энэ функцийг процессын хуваарь гэж нэрлэдэг.

Процессын хуваарийн дагуу үйлдлийн систем дараах ажлуудыг гүйцэтгэдэг:

- i.Процессуудын төлөв байдлыг хянах
- ii.Процессорыг процессуудад хуваарилах
- iii.Процессорыг процессуудад хуваарилах.

Эдгээр бүх процессыг үйлдлийн систем нь процесс бүрт хуваарилагдсан ID (PID)-ийн хамт процессын хүснэгтэд хадгалдаг бөгөөд энэ нь бүгдийг хянахад тусалдаг. Мөн тэдгээрийн одоогийн төлөвийг хянахын тулд үйлдлийн систем нь процессын хяналтын блок (ПХБ) ашигладаг. Процессын төлөв өөрчлөгдөх үед мэдээлэл нь хяналтын блокт шинэчлэгддэг.

Гэхдээ үйлдлийн систем эдгээр үйл явцыг төлөвлөхдөө ямар шалгуурыг ашигладаг вэ? Аль процессыг эхлээд CPU-ийн нөөцөөр хуваарилах ёстой вэ? Үйлдлийн систем нь хамгийн бага цаг зарцуулдаг процессыг гүйцэтгэх эсвэл процессор руу нэвтрэхийг хүссэн дарааллын дагуу гүйцэтгэх гэх мэт эдгээр шийдвэрийг гаргах олон арга зам байдаг. Ийм нэг чухал шалгуур бол үйл явцын тэргүүлэх чиглэл юм.

Үйлдлийн систем дэх тэргүүлэх төлөвлөлтийн хувьд процессуудыг тэргүүлэх ач холбогдлоор нь үндэслэн гүйцэтгэдэг. Өндөр ач холбогдол бүхий ажил/процессыг эхлээд гүйцэтгэнэ. Мэдээжийн хэрэг, та үйл явцын тэргүүлэх чиглэлийг хэрхэн шийддэгийг мэдэхийг хүсч магадгүй юм.

Процессын тэргүүлэх чиглэл нь зарим хүчин зүйлээс хамаарна, тухайлбал:

Хугацааны хязгаар

Процессын санах ойд тавигдах шаардлага

Дундаж оролт/гаралтын дундаж CPU-ийн тэсрэх хугацаатай харьцуулсан харьцаа
Үйл явц/ажлын тэргүүлэх чиглэлийг тодорхойлох олон хүчин зүйл байж болно. Энэ тэргүүлэх чиглэлийг төлөвлөгч процессуудад хуваарилдаг.

Процессын эдгээр тэргүүлэх чиглэлүүд нь 0-ээс 7 хүртэл, эсвэл 0-ээс 4095 хүртэлх тогтмол мужид энгийн бүхэл тоогоор илэрхийлэгддэг. Эдгээр тоо нь системийн төрлөөс хамаарна.

Одоо бид ажил бүрийг хуваарийн дагуу тэргүүлэх ач холбогдол өгдөг гэдгийг мэдэж байгаа тул бид ажил, төрлөөр нь урагшлах боломжтой.

3. Санах ой дахь процесс-Process in memory

Санах ойд боловсруулах (PIM) гэдэг нь санах ой болон төв боловсруулах нэгж (CPU) хооронд өгөгдөл дамжуулах хэрэгцээг багасгаж, өгөгдөл боловсруулах ажлыг санах ойн дэд систем дотор шууд гүйцэтгэдэг тооцоолох парадигмыг хэлдэг. Уламжлалт тооцоолох системүүд нь өгөгдлийг санах ойноос CPU руу шилжүүлж боловсруулж, дараа нь үр дүнг санах ой руу илгээдэг. PIM нь өгөгдлийн хөдөлгөөнийг багасгаж, санах ойн доторх өгөгдөл болон тооцооллын ойролцоо байдлыг ашиглах замаар тооцооллын үр ашгийг дээшлүүлэх зорилготой юм.

Программыг гүйцэтгэх үед түүнийг үндсэн санах ойд ачаалж, тодорхой хэмжээний зайг хуваарилдаг. Энэ зай нь програмын код, өгөгдөл, стекийг хадгалахад ашиглагддаг.

Санах ой дахь үйл явц:

Өгөгдөл хадгалах: Өгөгдлийг анх санах ойн дэд системд хадгалдаг бөгөөд үүнд RAM (санамсаргүй нэвтрэх санах ой) эсвэл санах ойд боловсруулах (PIM) төхөөрөмж гэх мэт PIM-д зориулагдсан тусгай санах ойн бүтцийг багтаадаг.

Өгөгдлийн ойролцоо тооцоолол: Тооцооллын үйлдлүүдийг боловсруулахад зориулж CPU-д шилжүүлэхийн оронд өгөгдөл хадгалагдаж буй санах ойн дотор эсвэл шууд хийгддэг. Энэ нь санах ойн элементүүд дээр шууд арифметик, логик эсвэл бусад үйлдлүүдийг гүйцэтгэхийг хэлнэ.

Өгөгдлийн давтамжийг багасгах: PIM нь санах ой болон CPU хооронд их хэмжээний өгөгдлийг зөөх хэрэгцээг эрс багасгаж, өгөгдөл дамжуулах явцад ихэвчлэн зарцуулдаг цаг, эрчим хүчийг хэмнэдэг.

Зэрэгцээ боловсруулалт: PIM нь өгөгдлийг санах ойд шууд зэрэгцүүлэн боловсруулах боломжийг олгодог бөгөөд энэ нь өөр өөр санах ойн нэгжүүд дээр олон үйлдлийг зэрэг гүйцэтгэх боломжийг олгодог.

Өгөгдлийн хувиргалт: Өгөгдлийг санах ойн доторх газар дээр нь хувиргах эсвэл дүн шинжилгээ хийх боломжтой бөгөөд үр дүн эсвэл завсрын өгөгдлийг санах ойд шууд үүсгэж болно.

Процесс ажиллаж дуусахад санах ойн зайг чөлөөлдөг. Дараа нь үйлдлийн систем нь энэ санах ойн зайг өөр процесст хуваарилладаг.

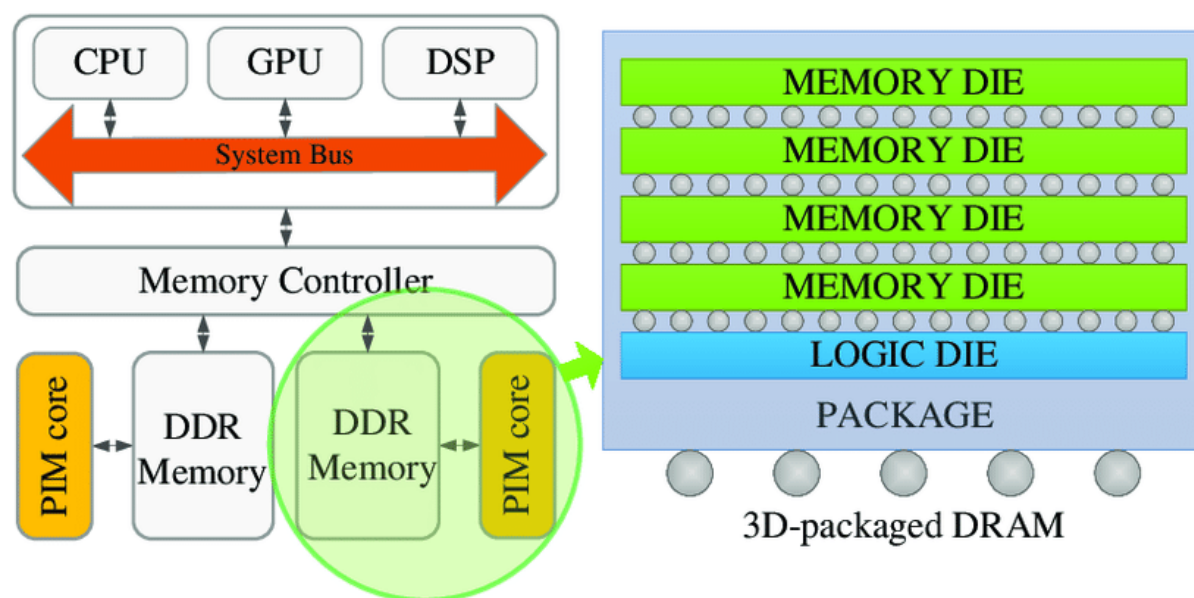
Санах ойн програмчлалын техник дэх үйл явц

Санах ойн програмчлалд процессыг хэрэгжүүлэхэд ашиглаж болох хэд хэдэн арга техник байдаг. Хамгийн түгээмэл техникүүдийн зарим нь:

Санах ойн мэдээллийн сан: Санах ойн мэдээллийн сан нь бүх өгөгдлийг санах ойд хадгалдаг. Энэ нь тэдэнд асуулга маш хурдан гүйцэтгэх боломжийг олгодог.

Өгөгдлийн бүтэц: Өгөгдлийн бүтцийг санах ойд өгөгдлийг хадгалах, хандах, удирдахад үр ашигтай байдлаар ашиглах боломжтой. Жишээлбэл, хэш хүснэгтүүд нь түлхүүр-утга хосыг хурдан хайх боломжтой байдлаар хадгалахад ашиглаж болно.

Алгоритмууд: Алгоритмуудыг санах ой дахь өгөгдөл дээр ажиллахын тулд дискний оролт гаралтын хэрэгцээг багасгахаар зохион бүтээж болно. Жишээлбэл, Quicksort нь санах ойд байгаа зүйлсийн жагсаалтыг дискэнд бичих шаардлагагүйгээр эрэмбэлэх алгоритм юм.



Санах ойн процессийн боловсруулалтын ашиг тус нь хэд хэдэн давуу талтай:

Хурд: Санах ой дахь өгөгдөлд хандах нь дискэн дээрх өгөгдөлд хандахаас хамаагүй хурдан байдаг. Учир нь санах ой нь дискнээс хамаагүй хурдан байдаг.

Өргөтгөх чадвар: Санах ойд өгөгдөл боловсруулах нь том өгөгдлийн багцыг зохицуулахад хялбар байдаг. Учир нь санах ойг компьютерийн системд суулгахад дискний санах ойгоос илүү амар учраас өгөгдөл боловсруулах нь дискэн дээрх өгөгдлийг боловсруулахаас хамаагүй хялбар юм. Учир нь дискний оролт гаралтыг удирдах шаардлагагүй байдаг.

Зэрэгцээ боловсруулалт ба РІМ:

Processing In Memory-ийн гол давуу талуудын нэг нь зэрэгцээ боловсруулалт хийх чадвар юм. Уламжлалт тооцооллын архитектурууд нь санах ой болон CPU хоёрын хоорондох өгөгдлийн хөдөлгөөний улмаас өгөгдлийг дараалан боловсруулах хэрэгцээ шаардлагаас байнга зовж байдаг. Үүний эсрэгээр РІМ нь санах ойн доторх өөр өөр өгөгдлийн элементүүдийг зэрэгцүүлэн боловсруулах боломжийг олгодог бөгөөд энэ нь тооцооллын үр ашгийг эрс сайжруулдаг.

Үр ашигтай мэдээллийн хөдөлгөөн:

РІМ нь CPU болон санах ойн хоорондох өгөгдлийн хөдөлгөөнийг эрс багасгадаг. Өгөгдлийг нааш цааш дамжуулахын оронд тооцоолол нь санах ойн дотор шууд хийгддэг. Энэ нь их хэмжээний өгөгдлийг зөөхтэй холбоотой эрчим хүчний хэрэглээ болон хоцролтыг багасгаж, үйл явцыг илүү үр дүнтэй болгодог.

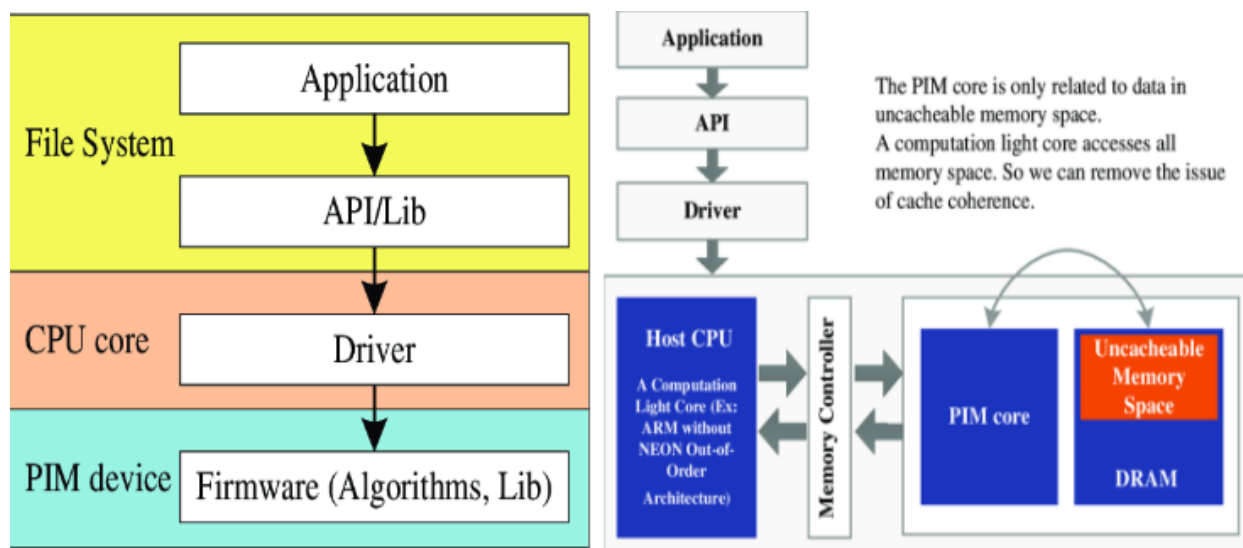
РІМ-ийн хэрэглээ:

РІМ нь янз бүрийн хэрэглээнд, ялангуяа том хэмжээний өгөгдөл боловсруулах, хиймэл оюун ухаан (AI), машин сургалт (ML), аналитик зэрэгт ихээхэн ашиг тустай. Матрицын үйлдлүүд, хэв маягийг таних, өгөгдөл шүүх зэрэг ажлуудыг РІМ ашиглан мэдэгдэхүйц хурдасгаж, илүү хурдан, илүү үр дүнтэй боловсруулалт хийхэд хүргэдэг.

Гэсэн хэдий ч санах ойн програмчлалын процесс нь зарим сул талуудтай:

Санах ойн шаардлага: Санах ойн програмчлалын процесс нь дискэн дээр суурилсан програмчлалаас илүү их санах ой шаарддаг учир нь програмыг бүхэлд нь санах ойд нэг дор ачаалах ёстой.

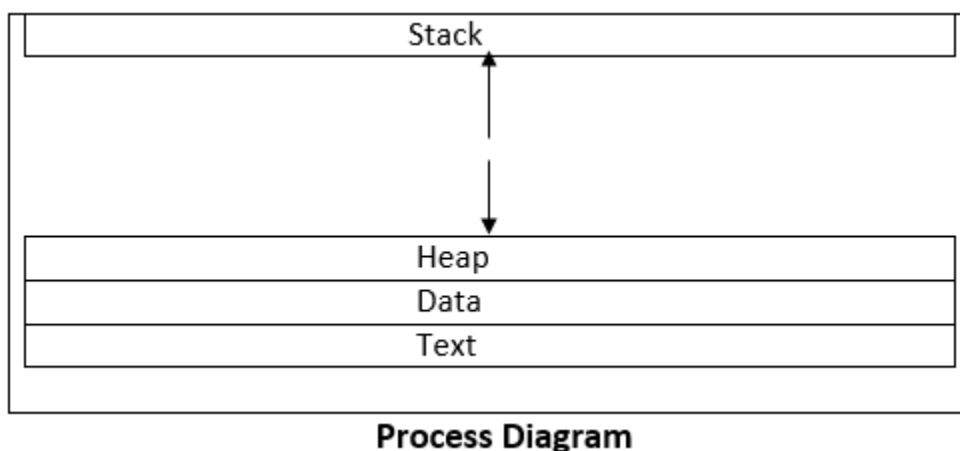
Эрчим хүчний хэрэглээ: Санах ойн програмчлалын процесс нь дискэн дээр суурилсан программчлалаас илүү их эрчим хүч зарцуулдаг. Учир нь санах ой нь дискнээс илүү тогтворгүй хадгалах хэрэгсэл юм.



4. Процессийн төлвүүд - амьдралын цикл

Процесс гэдэг нь одоо хэрэгжиж байгаа програмыг хэлнэ. Процессын тухай ойлголт нь үйлдлийн системийн үйл ажиллагаанд чухал байр суурь эзэлдэг бөгөөд энэ нь системд олон програмын гүйцэтгэлийг нэгэн зэрэг удирдах боломжийг олгодог. Програмыг ажиллуулах үед үйлдлийн систем нь түүнд зориулсан процесс үүсгэдэг бөгөөд үүнд өвөрмөц процесс танигч, санах ойн зай болон түүнийг хэрэгжүүлэхэд шаардлагатай бусад нөөцүүд багтдаг.

Үйлдлийн систем дэх үйл явцын амьдралын мөчлөг нь процессыг гүйцэтгэх явцад дамждаг янз бүрийн төлвийг хэлнэ. Процессын амьдралын мөчлөгийг таван үндсэн төлөвт хувааж болно.



Амьдралын цикл:

Хэрэглэгч ямар нэгэн процессыг гүйцэтгэхийг оролдох бүрт дуусгавар болох хүртэл олон үе шат дамждаг. Гэхдээ эдгээр үе шатууд нь үйлдлийн системээс хамаарч өөр өөр байж болдог тухайлбал хоёр, тав, долоон төлөвийн загвар гэх мэт нийтлэг үйл явцын амьдралын мөчлөгүүд байдаг.

Шинэ: Энэ төлөвт процессыг үүсгэж байгаа бөгөөд түүний нөөцийг үйлдлийн системээр хуваарилж байна.

Бэлэн төлөв: Үйлдлийн системд бэлэн төлөв нь үйлдлийн системийн хуваарьлагчаар процессорт хуваарилагдахыг хүлээж буй процессын төлөв юм. Энэ төлөвт процесс нь үндсэн санах ойд ачаалагдсан бөгөөд ажиллахад бэлэн байгаа боловч CPU бэлэн болтол хүлээж байна.

Процесс бэлэн байдалд байх үед ихэвчлэн гүйцэтгэхэд бэлэн байгаа бусад процессуудын дараалалд хүлээж байдаг. Үйлдлийн системийн хуваарь гаргагч нь тэргүүлэх ач холбогдол, бэлэн нөөц, хуваарийн алгоритм зэрэг олон хүчин зүйл дээр үндэслэн дараа нь аль процессыг гүйцэтгэх ёстойг тодорхойлдог.

Үйлдлийн системийн хуваарьлагч бэлэн дарааллаас процессыг сонгосны дараа процесс нь CPU дээр ажиллаж байгаа төлөв рүү шилждэг.

Ажиллаж байна: Энэ төлөвт процессыг процессор идэвхтэй гүйцэтгэж байна. Процесс нь даалгавраа биелүүлэхийн тулд CPU-г ашиглаж байна гэсэн үг.

Хүлээгдэж буй төлөв: Хүлээгдэж буй төлөв буюу блоклогдсон төлөв нь ямар нэгэн үйл явдал гарахыг хүлээж байгаа эсвэл үйлдлийн системээс түр зогсоосон учраас процесс одоогоор ажиллах боломжгүй байгаа төлөв юм. Процесс хүлээгдэж буй төлөвт орох үед процессор ажиллаж байгаа процессуудын дарааллаас хасагдаж, хүлээгдэж буй процессуудын дараалал руу шилждэг.

Хүлээх төлөвт байх үед хүлээж байсан үйл явдал болох хүртэл процесс идэвхгүй хэвээр байна. Үйл явдал тохиолдсоны дараа процессыг бэлэн байдалд шилжүүлж, үйлдлийн системээр гүйцэтгэхээр төлөвлөдөг.

Дуусгах төлөв: Дууссан төлөв нь үйл явцын амьдралын мөчлөгийн эцсийн төлөв юм. Процесс гүйцэтгэлээ дуусгасан эсвэл үйлдлийн системээр дуусгавар болсон үед энэ нь дуусгавар болсон төлөвт ордог. Энэ төлөвт санах ой, CPU-ийн цаг гэх мэт процессын нөөцийг бусад процессуудад ашиглахын тулд системд буцааж өгдөг.

Процесс дуусгавар болсон төлөвт орсны дараа түүнийг үйлдлийн системээр гүйцэтгэхээр төлөвлөх боломжгүй. Гэсэн хэдий ч, процессын гаралтын төлөвийг (процесс амжилттай дууссан эсэхийг харуулдаг утга) системийн дуудлагыг ашиглан эх процессоос авч болно.

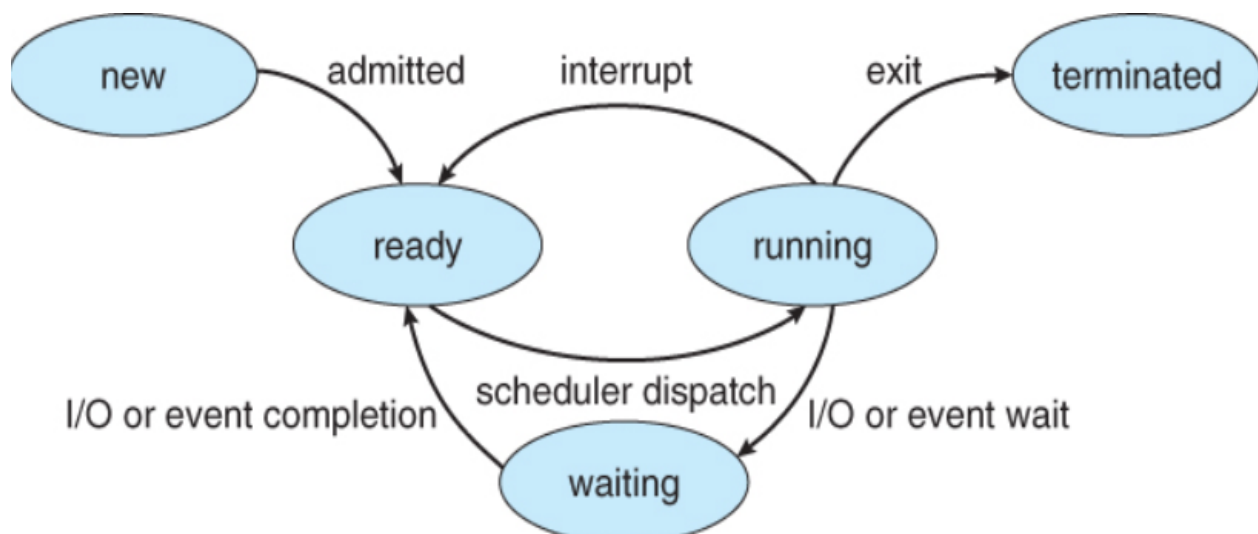
Дууссан төлөв нь процессын хувьд хэд хэдэн аргаар тохиолдож болно, үүнд:

Хэвийн гүйцэтгэл: Процесс нь ямар ч алдаагүйгээр гүйцэтгэлээ дуусгадаг.

Хэвийн бус дуусгавар: Сегментийн алдаа эсвэл зохицуулагдаагүй үл хамаарах зүйл зэрэг зарим алдаа эсвэл онцгой байдлын улмаас процесс дуусгавар болно.

Үйлдлийн системээр устгасан: Үйлдлийн систем нь хуваарилагдсан нөөцөө хэтрүүлсэн эсвэл аюулгүй байдлын зарим бодлогыг зөрчсөн гэх мэт зарим шалтгааны улмаас үйл явцыг зогсооно.

Ямар ч тохиолдолд процесс дуусгавар болсон төлөвт орсны дараа үйлдлийн систем дэх амьдралынхаа мөчлөгийг дуусгасан болно.



5.Процессын хэрэгжүүлэлт

Үйлдлийн системийн хичээлийн хүрээнд процессын хэрэгжилт гэдэг нь үйлдлийн систем процессуудыг хэрхэн удирдаж, гүйцэтгэж байгааг хэлнэ. Үйлдлийн систем дэх процесс нь компьютер дээр ажиллаж байгаа програмын жишээ юм. Үйлдлийн системүүд нь олон даалгаврыг зэрэг гүйцэтгэхийн тулд үйл явцыг үр ашигтайгаар зохицуулах

шаардлагатай бөгөөд энэ нь хэрэглэгчийн тасралтгүй туршлагыг бий болгодог. Үйлдлийн системд процессуудыг хэрэгжүүлэх нь дараах байдалтай байна.

1. Процесс үүсгэх:

Програмыг ачаалах: Хэрэглэгч програмыг эхлүүлэх үед үйлдлийн систем нь програмыг санах ойд ачаалдаг.

Процессын хяналтын блок (ПХБ): Үйлдлийн систем нь процесс бүрийн хувьд Процессын хяналтын блок (ПХБ) гэж нэрлэгддэг өгөгдлийн бүтцийг бий болгодог. ПХБ нь төлөв, програмын тоолуур, санах ойн заагч зэрэг процессын талаарх мэдээллийг агуулдаг.

2. Үйл явцын хуваарь:

Хуваарийн алгоритмууд: Үйлдлийн систем нь дараачийн аль процессыг гүйцэтгэхийг тодорхойлохын тулд хуваарийн алгоритмуудыг (эргэлдүүлэх, тэргүүлэх хуваарь гэх мэт) ашигладаг. Эдгээр алгоритмууд нь процессорыг процессор гүйцэтгэх дарааллыг удирддаг.

3. Процессын гүйцэтгэл:

CPU-ийн гүйцэтгэл: Үйлдлийн систем нь процессуудад CPU-ийн цагийг хуваарилж, зааварчилгаагаа биелүүлэх боломжийг олгодог.

Контекст шилжих: Үйлдлийн систем нь нэг процессоос нөгөө процесс руу шилжих үед контекст шилжүүлэгчийг гүйцэтгэдэг. Энэ нь одоогийн үйл явцын контекстийг (бүртгэл, програмын тоолуур гэх мэт) хадгалах, дараагийн гүйцэтгэх процессын контекстийг ачаалах явдал юм.

4. Процессын харилцаа холбоо ба синхрончлол:

Процесс хоорондын харилцаа холбоо (IPC): Процессууд нь ихэвчлэн хоорондоо харилцах шаардлагатай болдог. Үйлдлийн системүүд нь мессеж дамжуулах, хуваалцсан санах ой зэрэг IPC-ийн механизмуудыг хангадаг.

Синхрончлол: Зөрчилдөөнөөс зайлсхийхийн тулд процессууд үйл ажиллагаагаа синхрончлох шаардлагатай байж магадгүй. Үйлдлийн системүүд нь семафор, мутекс зэрэг синхрончлолын механизмуудыг хангадаг.

5. Үйл явцыг дуусгавар болгох:

Ердийн дуусгавар: Процессууд даалгавраа дуусгасны дараа дуусгавар болж болно.

Хэвийн бус дуусгавар: Процессууд алдааны улмаас дуусгавар болж магадгүй. Үйлдлийн систем нь зохих цэвэрлэгээг баталгаажуулж, нөөцийг гаргаж, үйл явцын статусыг шинэчилдэг.

6. Санах ойн менежмент:

Санах ойн хуваарилалт: Үйлдлийн системүүд нь процессуудын санах ойн хуваарилалт, хуваарилалтыг удирддаг.

Виртуал санах ой: Үйлдлийн системүүд нь ихэвчлэн виртуал санах ойг хэрэгжүүлдэг бөгөөд энэ нь RAM болон дискний санах ойн хооронд өгөгдөл солилцох замаар процессуудад физикээс илүү санах ой ашиглах боломжийг олгодог.

7. Нөөцийн менежмент:

Файлын системийн хандалт: Процессууд нь файлаас унших эсвэл файл руу бичих шаардлагатай байж магадгүй. Үйлдлийн систем нь файлын системийн хандалтыг удирдаж, аюулгүй байдлыг хангаж, зөрчилдөөнөөс урьдчилан сэргийлдэг.

Төхөөрөмжийн удирдлага: Процессууд нь техник хангамжийн төхөөрөмжтэй харилцаж болно. Үйлдлийн систем нь төхөөрөмжийн драйверуудыг хангаж, зөрчлөөс урьдчилан сэргийлэхийн тулд төхөөрөмжийн хандалтыг удирддаг.

8. Алдаа засах, сэргээх:

Алдаа илрүүлэх: Үйлдлийн систем нь хууль бус зааварчилгаа, санах ойн хандалтын зөрчил зэрэг процессуудын алдааг илрүүлдэг.

Алдаа сэргээх: Үйлдлийн систем нь алдаатай үйл явцыг зогсоох, холбогдох нөөцийг чөлөөлөх зэрэг зохих арга хэмжээг авдаг.

Үйлдлийн систем дэх үйл явцын хэрэгжилтийг ойлгох нь орчин үеийн тооцооллын системд олон үйлдэл хийх, нөөцийг үр ашигтай удирдах үндэс суурийг бүрдүүлдэг тул компьютерийн эрдэмтэн, инженерүүдийн хувьд маш чухал юм. Өөр өөр үйлдлийн системүүд эдгээр ойлголтыг өвөрмөц байдлаар хэрэгжүүлж болох ч үндсэн зарчмууд нь тогтвортой хэвээр байна.

Процессын бүтэцтэй үйл ажиллагааны хамгийн доод давхарга систем нь тасалдал болон хуваарийг зохицуулдаг. Тэр давхаргын дээгүүр дараалсан процессууд юм.

1. Техник хангамжийн стекийн програмын тоолуур гэх мэт.
2. Техник хангамж нь тасалдлын вектороос шинэ програмын тоолуурыг ачаална.
3. Ассемблей хэлний процедур нь региструудийг хадгалдаг.
4. Ассемблей хэлний процедур нь шинэ стекийг тохируулдаг.
5. С тасалдлын үйлчилгээний ажил (ихэвчлэн оролтыг уншиж, буфер болгодог).
6. Төлөвлөгч нь дараа нь аль процессыг ажиллуулахыг шийддэг.
7. С процедур нь угсралтын код руу буцдаг.
8. Ассемблей хэлний процедур нь одоогийн шинэ процессыг эхлүүлнэ

Төлөвлөлтийн алгоритмууд

Batch

- Interactive
- Real time

Төлөвлөлтийн алгоритмын зорилго

Бүх системүүд Fairness процесс бүрд CPU-ийн бодит хувь хэмжээг өгдөг Бодлогын хэрэгжилт - заасан бодлого хэрэгжиж байгааг хардаг Тэнцвэртэй байх - системийн бүх хэсгийг завгүй байлгах Багц системийн хүчин чадал - нэг цагийн ажлын хэмжээг нэмэгдүүлэх - Х Эргэлтийн хугацаа - хоорондох хугацааг багасгах Хэрэглээ ба дуусгавар CPU-ийн ашиглалт - CPU-ийг үргэлж завгүй байлгах Интерактив системүүд Хариу өгөх хугацаа - хүсэлтэд хурдан хариу өгөх Пропорциональ байдал - хэрэглэгчийн хүлээлтийг хангах Бодит цагийн системүүд Уулзалтын эцсийн хугацаа - өгөгдөл алдагдахаас зайлсхийх Урьдчилан таамаглах чадвар - мультимедиа системийн чанарын доройтлоос зайлсхийх

Batch систем дэх төлөвлөлт

- First-come first-served • Shortest job first • Shortest remaining Time next

Scheduling in Interactive Systems

- Round-robin scheduling • Priority scheduling • Multiple queues • Shortest process next • Guaranteed scheduling • Lottery scheduling • Fair-share scheduling

Процессын дөрвөн хэрэгжүүлэлт

1. First come first serve - хамгийн эхэнд ирсэн процесс хамгийн түрүүнд ажиллана
2. Shortest job first - хамгийн бага алхамтай процессоос эхэлж ажиллана.
3. Round robin - Тодорхой агшин тутамд процессыг солих арга.
4. Priority - давуу эрхийн арга. Хэрэглэгч өөрийн хүссэнээр процессуудаа хувиарлаж болно.

1. First come first served

Process	Service time T_s	Turnaround time T_r	Waiting time T_w
A	10	10	0
B	1	11	10
C	3	14	11
D	4	18	14
Average		13.25	8.75
$T_r/T_w = 1.51$		$T_r/T_s = 5.29$	

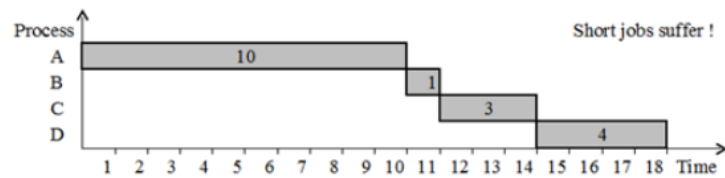


Figure 3: An example to First-Come First-Served.

2. Shortest job first

Process	Service time T_s	Turnaround time T_r	Waiting time T_w
A	10	18	8
B	1	1	0
C	3	4	1
D	4	8	4
Average			
$T_r/T_w = 2.38$		$T_r/T_s = 1.53$	

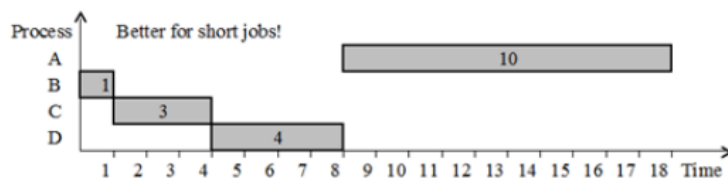


Figure 5: An example to Shortest Job First.

3. Round robin

Process	Service time T_s	Turnaround time T_r	Waiting time T_w
A	10	18	8
B	1	2	1
C	3	9	6
D	4	12	8
Average		10.3	5.75
$T_r/T_w = 1.78$		$T_r/T_s = 2.45$	

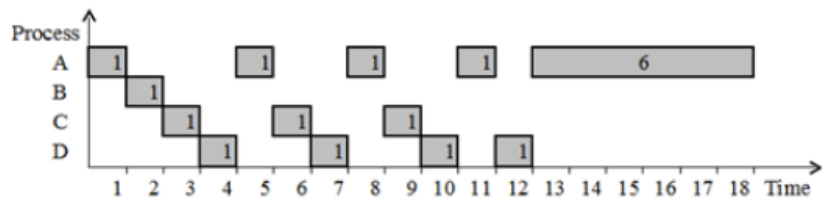


Figure 8: An example to Round Robin.

4. Priority

Process	Priority	Service time T_s	Turnaround time T_r	Waiting time T_w
A	4	10	18	8
B	3	1	8	7
C	2	3	7	4
D	1	4	4	0
Average		4.5	9.25	4.75
		$T_r/T_w = 1.95$	$T_r/T_s = 3.28$	

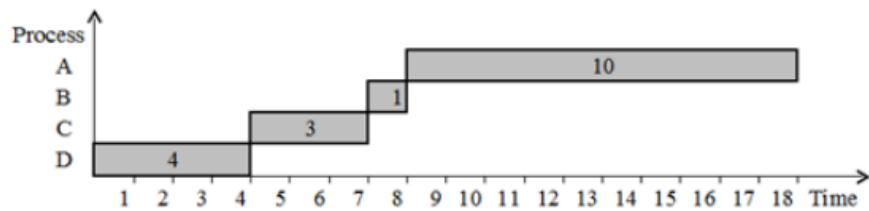


Figure 10: An example to Priority-based Scheduling.

6.Процесийн үзүүлэлтүүд

Процессын үзүүлэлтүүд нь ихэвчлэн компьютерийн архитектур эсвэл дижитал системийн дизайны хүрээнд тодорхой үйл явцын нарийвчилсан шинж чанар, шаардлагыг илэрхийлдэг. Таны өгсөн жишээнүүдийг задалъя:

1. MIPS (Сая дахь зааварчилгаа):

Тодорхойлолт: MIPS нь компьютерийн боловсруулалтын хурдны хэмжүүр юм. Энэ нь CPU-ийн нэг секундын дотор хэдэн сая зааварчилгааг гүйцэтгэхийг илэрхийлдэг.

Жишээ тодорхойлолт: Компьютерийн процессор нь MIPS үнэлгээ 2 байж болох бөгөөд энэ нь секундэд 2 сая заавар гүйцэтгэх чадвартай гэсэн үг юм.

2. Цаг (Цагийн хурд):

Тодорхойлолт: Герцээр (Гц) хэмжигдэх цагийн хурд нь CPU секундэд хэдэн цикл гүйцэтгэж болохыг илэрхийлдэг. Энэ нь процессорын хурдны үндсэн үзүүлэлт юм.

Жишээ тодорхойлолт: Процессор нь 3.5 GHz (секундэд 3.5 тэрбум цикл) цагийн хурдтай байж болох бөгөөд энэ нь түүний секунд тутамд 3.5 тэрбум үйлдэл хийх чадвартайг харуулж байна.

3. CPI (Заавар бүрийн мөчлөг):

Тодорхойлолт: CPI нь нэг тушаалыг гүйцэтгэхэд шаардагдах цагийн мөчлөгийн дундаж тоог хэмждэг. Энэ нь процессор зааварчилгааг хэр үр дүнтэй гүйцэтгэдэг талаар ойлголт өгдөг.

Жишээ тодорхойлолт: Хэрэв процессорын CPI нь дунджаар 2 байвал нэг командыг гүйцэтгэхэд 2 цагийн цикл шаардлагатай.

Компьютерийн системд эдгээр үзүүлэлтүүд нь янз бүрийн процессоруудын гүйцэтгэлийг ойлгох, харьцуулахад маш чухал юм. Өндөр MIPS утга, илүү хурдан цагийн хурд, бага CPI нь ерөнхийдөө илүү хурдан, илүү үр ашигтай процессорыг илтгэнэ. Гэсэн хэдий ч эдгээр үзүүлэлтүүд нь түүхийг бүхэлд нь хэлж чадахгүй гэдгийг анхаарах нь чухал юм. Процессорын бодит гүйцэтгэл нь түүний архитектур, дамжуулах хоолой, параллелизм болон бусад хүчин зүйлээс хамаарна.

Процессын техникийн үзүүлэлтүүд нь компьютерийн архитектор, техник хангамжийн инженер, програм хангамж хөгжүүлэгчдийн хувьд чухал ач холбогдолтой бөгөөд тэдгээр нь системийг зохион бүтээх, програм хангамжийг оновчтой болгох, янз бүрийн техник хангамжийн бүрэлдэхүүн хэсгүүдийн нийцтэй байдлыг хангахад тусалдаг. Процессорын үзүүлэлтийг хэрэглэж байгаа зааврын тоогоор хэмждэг.

- MIPS - Секунд дэх сая заавар (Instructions Per Second), 1 секундэд ажиллах сая (1000000) зааврын тоо
- Clock : Клокийн тоо нь заавраас хамаардаг.

Клокун тооны урвуу хэмжигдэхүүнийг клокун давтамж гэж нэрлэдэг.

Клокун давтамж нь PC үзүүлэлтийг хэмждэг индекс болдог.

Процессорын үндсэн үйлдэл клокун тусламжтай гүйцэтгэгддэг.

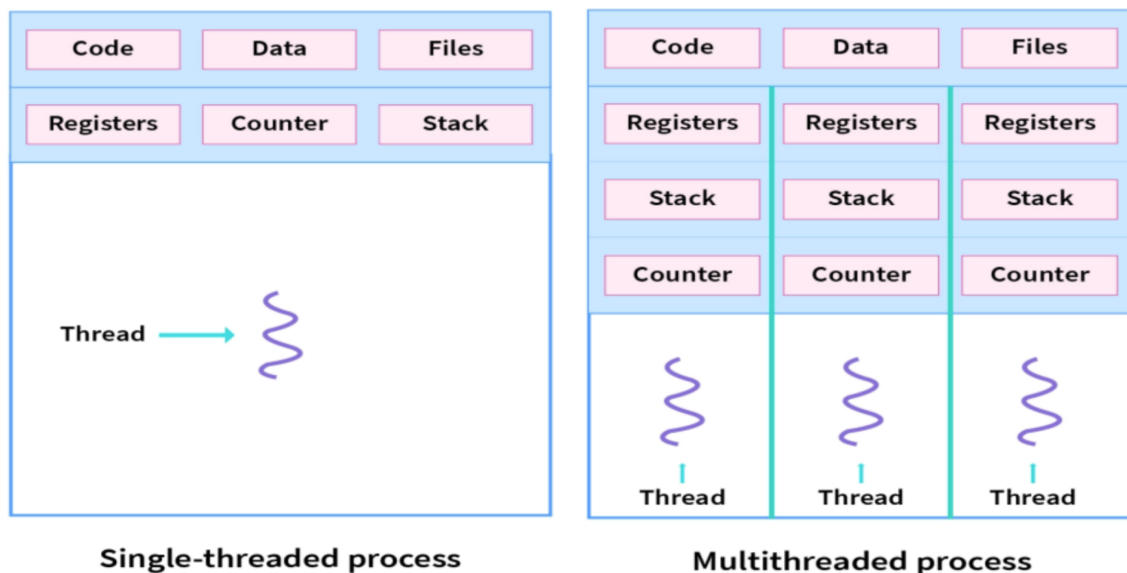
- CPI - Заавар дах цикл (тактын дохио) (Cycles Per Instruction), 1 зааврыг ажиллуулахад шаардагдах клокун тоо (Cycles Per Instruction)

7. Салбар Процесс

Thread нь процесс доторх даалгавруудын дараалсан урсгал юм. Үйлдлийн систем дэх thread нь ижил эсвэл өөр төрлийн байж болно. Thread нь програмуудын гүйцэтгэлийг нэмэгдүүлэхэд ашиглагддаг.

Thread бүр өөрийн програмын тоолуур, стек, регистрийн багцтай. Гэхдээ нэг процессын хэлхээнүүд нь ижил код, өгөгдөл/файлыг хуваалцаж болно. Thread нь нийтлэг нөөцийг хуваалцдаг тул хөнгөн процесс гэж нэрлэгддэг.

Эхэн үеийн CPU мэдээлэл боловсруулах чадамж муутай мөн компьютерт холбогдсон төхөөрөмжүүдтэйгээ харилцан ажиллах шаардлагатай ч тэдгээр нь маш удаан байсан. Шууд холбогдсон оролтын төхөөрөмжөөс оруулсан өгөгдлийг CPU боловсруулахын тулд ажиллаж буй процессыг зогсоодог байсан. Энэ нь нэлээд үр дүн муутай арга байв. Олон програмыг ажиллуулах горимыг (multiprogramming) анх LEO III компьютерийн системд ашигласан.



Зураг 2.1 процессын харьцуулсан зураг

Компьютерийн системд процесс нь хоёр шинж чанарын нийлмэл цогц гэж ойлгож болно. Эдгээрийн шинж чанар нь :

- Нөөцийг эзэмшигч
- Ажиллагч юм

Процесс нь өөрийн нөөцийг эзэмшигч шинж чанараар эзэмшиж буй бүх нөөцийн хэлбэр, байршил гэх мэт параметрыг дүрсэлж байдаг. Харин хэрхэн ажиллаж байгаа дүр зураг буюу регистрийн утгууд, төлөв, зэрэглэл гэх мэт параметруудыг ажиллагч шинж чанартаа хамааруулан авч үзээд процессын хуулбар /thread/ гэж нэрлэе.

Процессын хуулбар буюу хуулбар процессын мөн хөнгөн жингийн процесс ч /Lightweight process/ гэж нэрлэдэг. Өөрөөр хэлбэл процессын ажиллах явцад өөрчлөгдөж байдаг мэдээллийг л хуулбар гэж нэрлэнэ. Үйлдлийн системд процесс нэг буюу хэд хэдэн хуулбартай байж болно. Өөрөөр хэлбэл нэг процесс нь хэд хэдэн ажиллах шинж чанартай байж болно гэсэн үг.

Процессын орон зай дахь хуулбарын загвар

Хуулбар процессыг ашиглах нь процессыг дангаар нь ашигласнаас давуу талуудтай. Зарим давуу талыг дурьдвал ;

- Хуулбар шинээр үүсгэхэд процесс үүсгэхээс бага хугацаа зарцуулна.
- Хуулбарыг устгах , хаахад процессыг устгах , хаахаас бага хугацаа зарцуулна.
- Хуулбар хооронд шилжих нь процесс хооронд шилжихээс хурдан байна.
- Нэг процессын орон зайд байгаа хуулбарууд хоорондоо харьцах үйлдэлд процесс хооронд харьцсанаас бага хугацаа зарцуулна гэх мэт.

Мөн нэг процессын орон зайд байгаа хуулбарууд орон зайг нь эзэмших байгаа процесс нь түр зогсвол дагаж түр зогсох ба харин процесс нь уствал уг процессын хуулбарууд нь Мөн устана.

Процесс болон салбар процесс роцессууд нь гүйцэтгэх үндсэн нэгжийн хувьд даалгавруудыг жижиг, удирдах боломжтой хэлхээ болгон хуваах замаар компьютерт олон ажлыг үр дүнтэй гүйцэтгэх боломжийг олгодог. Процесс бүр нь бие даасан байдлаар ажилладаг бөгөөд компьютерийн орчинд тусгаарлалт, аюулгүй байдлын мэдрэмжийг бий болгодог. Процессууд нь системийн нөөцийг үр ашигтай ашиглах боломжийг олгож, янз бүрийн програм, даалгавруудыг жигд гүйцэтгэх боломжийг олгодог. Салбарлах нь процессыг олон урсгалд хуваах боломжийг олгодог бөгөөд тус бүр нь өөр өөр ажлыг нэгэн зэрэг гүйцэтгэх чадвартай. Энэхүү зэрэгцээ гүйцэтгэл нь системийн ерөнхий гүйцэтгэлийг сайжруулж, нарийн төвөгтэй ажиллагаа, тооцооллыг илүү хурдан гүйцэтгэх чадвартай болгодог. Салбарын үйл явц нь даалгавруудыг нэгэн зэрэг хувааж, боловсруулах боломжтой хувилбаруудад маш чухал бөгөөд тэдгээрийг дуусгахад шаардагдах хугацааг эрс багасгадаг.