

## Лекция №8.

### Тақырыбы: Енгізу-шығару жүйесін басқару. Синхронды және асинхронды енгізу-шығару.

Енгізу-шығару жүйесін басқару – ол ОЖ-гі негізгі функцияларының бірі және ОЖ жобалауының ең қиын аймақтарының бірі.

Енгізу-шығару құрылғыларының көптігінен және олардың түрлі табиғаттарынан қиыншылықтар туады. Сондықтан ОЖ-ні құрушылардың арасында оңай есеп тұрған жоқ. Олар енгізу-шығару құрылғыларының тиімді басқаруды ғана емес қолданба программистер құрылғылардың түріне қарамастан деректерді оңай оқып, сақтау үшін ыңғайлы, тиімді виртуальды интерфейсті құру қажет.

Бір модельге бірнеше түрлі құрылғыларды біріктіретін жүйелік енгізу-шығару әмбебап болу қажет. Біріншіден ол бар құрылғылардың қажеттіліктерін ескеру қажет. Екіншіден бірнеше параллель орындалып жатқан есептерге енгізу-шығару құрылғыларына қатынауды ескеру қажет.

Қолданбалы программалар өздері енгізу-шығару құрылғылары тікелей байланысалмайды. Енгізу-шығару сұрауына керек мәндерді орнатып, енгізу-шығару супервизорына басқару беріледі.

Енгізу-шығаруды басқару режимдері. Екі негізгі енгізу-шығару режимі бар: енгізу-шығару құрылғыларының дайындығын сұрату алмасу режимі және үзілісі бар алмасу режимі.

Егер енгізу-шығару басқаруын орталық санашық атқарса онда ол басқару құрылғысына бір енгізу-шығару құрылғысының орындалуына команда береді. Енгізу-шығару құрылғысы команданы орындап орталық санашыққа және сигналды басқаратын құрылғыға түсінікті сигнал жібереді.

Енгізу-шығару құрылғысының жылдамдылығы орталық санашықтың жылдамдылығынан төмен. Сондықтан дайындық сигналын көп күту керек, сигнал шыққанша драйвер еш нәрсе орындамайды. Сондықтан енгізу-шығару командасын беріп ол құрылғыны уақытша ұмытып басқа программаның орындалуына көшкен жөн. Ал дайындық сигналы шыққан кезде оны енгізу-шығару құрылғысының үзуге сұрау деп санаймыз.

Ортақ енгізу-шығару құрылғылары, құрылғыларды жалғау. Көп құрылғылар бірігіп қолдануды қолдамайды. Біріншіден ол тізбекті қатынайтын құрылғылар. Ондай құрылғылар жалғанған болуы мүмкін, яғни бір есептеу үрдіске беріледі. Бірақ ол есептеу үрдістерінің параллельді жұмыс жасауына мүмкіндік бермейді және енгізу-шығару құрылғыларының босау уақытын күтеді.

Параллельді орындалып жатқан есептер арасында енгізу-шығару құрылғыларын қолдануын ұйымдастыру үшін виртуальді құрылғылар түсінігі енгізіледі.

Виртуализация принципін қолдану есептеу жүйесінің тиімділігін жоғарлатуға мүмкіндік береді. Көбінесе виртуалды құрылғыға – спулинг терминін қолданады.

*Спулинг дегеніміз-* операциялайтын режимде құрылғылармен жұмыс иммитациясы.

Спулингтің негізгі жұмысы тізбекті қатынайтын құрылғыларының параллельді бөлінуінің көрінісін ұйымдастыру. Бірақ олар монопольді және жалғанған ретінде қолдану қажет.

Мысалы, егер біз бірнеше қолданбаның қорытындысын басуға жіберетін болсақ, онда біз түсініксіз қорытындыға ұшыраймыз. Бірақ әр есептеу үрдісіне виртуальді принтер беруге болады және ағындылар бірінші рет магнитті дискідегі арнайы спул файлына жіберіледі.

Содан соң виртуалды басу аяқталған соң қабылданған пұрсаттылықтармен және қызымет көрсету пәні бойынша спул файлдың мәні принтерге жіберіледі. Спул файлымен басқаратын жүйелік үрдіс – спулер деп аталады (spool-reader немесе spool-writer).

Программисті компьютерге орнатылған енгізу-шығару құрылғыларының параметрлері және олардың мүмкіншіліктері мүмкіншілік болғанша ойландырмау керек. Оған тек қолданатын енгізу-шығару құрылғыларының жалпы мүмкіншіліктері қажет. Мысалы, принтер басуды білу қажет, ал магниттік дискіде жинақтауыш берілген адрес бойынша (C-N-S координаты, цилиндр номері, бастиек номері және сектор номері) деректерді жазу және оқу қажет.

Бірақ программист магниттік дискімен жұмыс арасында тікелей адрестеуді қолданбайды. Ол файлдық жүйе деңгейімен жұмыс жасайды. Бұл жағдайда қандай жинақтауыш, қай типті және моделі компьютерде қолданылғаны файлдық жүйе құрушыларын тәуелді болмауы қажет.

### ***Енгізу-шығаруды диспетчерлеу. Еркін және тізбекті қатынас құру тәсілдері.***

Есептерді динамикалық жоспарлау үрдістерді жоспарлауға қарағанда актуалды, өйткені қорларды тиімді үлестіру әр жағдайда болады. Есептің динамикалық жоспарлауын *диспетчерлеу деп атайды*. Орындалып жатқан үрдістермен ағындар арасында ағымдағы қорларды үлестіруден жоспарлау көп ұйымдастырылады.

Орындалуға дайын есептердің кезегінің тізімін дайындайтын көп диспетчерлеу пәндері белгілі. Қызмет көрсететін пәндер екі үлкен топқа бөлінеді – пұрсаттылықты және пұрсаттылықсыз.

Мультипрограммалық ОЖ-де көп есептер орындалған кезде деректерді оқуға және жазуға сұраныс көп ағындар болып өтуі мүмкін, сондықтан оларды орындау кезінде кезек пайда болады. Егер біз сұраныстарды олардың кезекке түскеніне байланысты орындайтын болсақ, онда магниттік дискінің секторларына кездейсоқ қатынайтын боламыз, онда деректерді жабуға біз көп уақыт жібереміз. Өйткені деректерді іздеп табуға уақытты азайту үшін сұраныстарды кезекке дұрыстап қоюға болады және олар тез орындалу үшін тиімді әдісті табу қажет. Ол проблеманы шешу үшін қажет тиімді жоспарлау пәнін табу қажет. Белгілі пәндерді қарастырайық, олар деректерді оқу/жазу операцияларына сұраныс кезегін қайта құрады.

### ***Енгізу-шығару ағындары. Енгізу-шығаруды қорғау тәсілдері***

Енгізу-шығару сұраныстары нақты белгіленген бірнеше оңдеу этаптарынан өтеді. Бұл этаптарда сұраныс құрылғыға бірқатпарлы драйвер арқылы басқарылатыны немесе көпқатпарлы драйвер арқылы өтуіне байланысты айырмашылықтары бар. Әрі қарай айырмашылықтарды оңдеу кезінде берілген енгізу-шығару типі – синхронды немесе асинхронды бойынша белгіленеді. Мысалы екі үрдіс жалпы оперативті жадының кеңістігінде жұмысын орындайды. Бұнда бөлінетін қорлар әр түрлі режимдерде орындалуы мүмкін, мысал екі үрдіс екі бөлек машинада болса, бірақ олар жалпы оперативті жадының өрісімен байланысуы мүмкін. Бұл жағдайда жадымен буферлеу жұмысын атқарған кезде проблема түзелуі мүмкін, өйткені әр машинаның өзінің оқу/жазу буферлеу механизмдері болады. Онда жағымсыз жағдай туады, физикалық жадының жағдайы оның реалды мәніне сәйкес келмейді. Сонымен қатар екі бөлек машинада жұмыс жасап отырған операциялық жүйеде проблема түзеледі.