ӨЗБЕКСТАН РЕСПУБЛИКАСЫ БАЙЛАНЫС, ИНФОРМАТИЗАЦИЯЛАЎ ҲӘМ ТЕЛЕКОММУНИКАЦИЯ ТЕХНОЛОГИЯЛАРЫ МӘМЛЕКЕТЛИК КОМИТЕТИ

ТАШКЕНТ ИНФОРМАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАРЫ УНИВЕРСИТЕТИ НӨКИС ФИЛИАЛЫ

Информациялық технологиялар кафедрасы

Компьютер инжиниринги факультетиниң
«Информатика ҳәм информациялық технологиялар» бағдарының
4-курс студенти Халиков Бабаназардың

ПИТКЕРИЎ ҚӘНИГЕЛИК ЖУМЫСЫ

Темасы: “ Сызықлы емес структурадағы мағлыўматлар тийкарында программаластырыўда мағлыўматлар базаларынан пайдаланыў ”

Илимий басшы: ______________________ **«Ўзбекистон почтаси» ААЖ Қарақалпақстан филиалы директоры Б.Машарипов**

______________________ **асс. А.Рысназаров**

Кафедра баслығы: ______________________ **т.и.к. Арзымбетов Т.З.**

НӨКИС - 2014 ж.

Мазмуны:

Кирисиў ……………….…………………………….……………………3

§1. Мағлыўматлар структуралары ҳәм олар үстинде орынланатуғын әмеллер ………………………..……..………5

§2. Сызықлы емес структуралы мағлыўматлар ҳәм оларды програмаластырыўда қолланыў ……………………...…….12

§3. Сызықлы емес структуралы мағлыўматлар ушын мағлыўматлар базаларын қолланыў ……………..……………26

Жуўмақлаў……………………………………………...…………39

Әдебиятлар

КИРИСИЎ

Жәмийетимизде информацион коммуникация технологияларының раўажланыўы бул заманагөй техникаларда оларды жумыс ислеўин тәмийнлеўши программаларды жаратыў, усы программалардың информациялық тәмийнлениўи мәселелерин шешиў ҳәм қоллана билиў көнликпелерин ийелеў кереклигин көрсетпекте.

Компьютердеги информациялық системалардың ўазыйпасы мағлыўматларды қайта ислеў болғанлығы, усы мағлыўматлардың бирдей структураларда болмаўын есапқа алсақ программаластырыўда бир тегис усылларды қолланыў мүмкин емеслиги көринеди.

Программластырыўшы ҳәзирги дәўирде программаластырыў технологияларының имканиятларынан пайдалана билиўи ҳәмде ҳәр-қыйлы структуралардағы мағлыўматларды программаластырыўда қолланғанда компьютер ядын ҳәм ислеў ўақытын тежеп пайдалана билиўи керек.

Мениң питкериў қәнигелик жумысым сызықлы емес структурадағы мағлыўматлар тийкарында программаластырыўда мағлыўматлар базаларынан пайдаланыўға арналған болып: кирисиўден, үш параграфтан, жуўмақлаўдан ҳәм әдебиятлардан турады.

Биринши параграфта программаластырыўда қолланылатуғын мағлыўматлар структуралары ҳәм олар үстинде орынланатуғын әмеллер үйрениледи.

Екинши параграфында сызықлы емес структуралы мағлыўматлар түрлери, олардың дүзилислери, уқсаслықлары, оларды белгили түрдеги мағлыўматлардың байланыслары көринисине келтириў ҳәм оларды программаластырыўда қолланыў қарастырылып өтиледи.

Үшинши параграфында сызықлы емес структуралы мағлыўматларды пайдаланып программаластырыўда мағлыўматлар базаларын қолланыўға арналып, сызықлы емес структуралар арасындағы байланысларды мағлыўматлар базалары таблицаларында сәўлелендириў мәселеси шешиледи. Ҳәмде анық мағлыўматлар тийкарында мысал алынып, олардың мағлыўматлар базасы веб серверлерде қолланылатуғын MySQL мағлыўматлар базасы таблицалары ушын дүзилип, программа коды PHP скриптлик тили жәрдеминде жазылыўы көрип өтиледи.

§1. Мағлыўматлар структуралары ҳәм олар үстинде орынланатуғын әмеллер.

Берилген мәселени компьютер жәрдеминде шешкенимизде, керекли информация ҳәр-қыйлы түрлердиң элементлериниң жыйындысы көринисине келтириў керек.

Программаластырыўдың ҳәр-бир системаларында қурамалы структураларды жәмлестирип пайдаланыў ушын мағлыўматлардың әпиўайы түрлери ҳәм қуралларынан пайдаланылады. Мағлыўматлардың әпиўайы түри дегенде пүтин, логикалық, ҳақыйқый ҳәм символлық түрлер саналады. Программаластырыўдың көплеген системаларының ишинде әпиўайы түрлерден бир түрдеги элементлерден турыўшы массивлер, хәр-түрли шекли сандағы жазыўлар ҳәм тағы басқа структураларды группалаўға болады.

Улыўма айтқанда мағлыўматлар түри дегенимизде мәнислер көплиги ҳәм олар үстинде орынланатуғын операциялар көплиги түсиниледи. Бундай операцияларға төмендегилерди мысал етип көрсетиўимизге болады.

- Қарап шығыў ушын бос көпликти киритиў;
- Көпликке элемент қосыў;
- Көпликтен элемент алып таслаў;
- Қарастырып атырған көплигимиздиң бослығын тексериў;
- Қарастырып атырған көплигимизде неше элемент барлығын билиў;

Компьютер орайлық процессордың системалық командалары менен анықланыўшы бөлек битлердиң алгоритмлеринен турыўшы мағлыўматлар менен операцияларды орынлап жумыс ислейди. Компьютер арқалы шешилиўши мәселелер ҳәмме ўақыт битлер көринисинде бола бермейди. Олар санлар, литерлик, текстлик, символлық ҳәмде қурамалырақ болған избе-изликлер, дизимлер ҳәм тереклер түўриндеги структуралар болыўы мүмкин. Мағлыўматлардың структураларының машина ядындағы көринисин есапқа алмаған жағдайда қарастырылыўын абстракт яки логикалық

структуралар деп атаймыз. Ал кери жағдайда ямаса мағлыўматлардың машина ядындағы көринисин «мағлыўматлардың физикалық структурасы» деп аталады. Буннан келип шыққан ҳалда есаплаў системаларында, логикалық басқыштан физикалық басқышқа өткизиўши ҳәмде керисинше өткизиўши процедуралар бар болады.

Мысалы логикалық басқышта еки өлшемли массивтиң элементине рухсат алыўда оның таблицадағы қатар ҳәм бағана номерин қорсетиў менен әмелге асырылады. Физикалық баскышта алдыннан белгили массив ушын ажыратылған яддағы адрестен адреслеў функциялары аркалы шақырылады. Солай етип мағлыўматлардың ҳәр-бир структурасын логикалық (абстракт)) ҳәм физикалық (конкрет) деп характерлеўге болады. Усы еки басқыш арасындағы операциялардың байланысы төмендеги сүўретте көрсетилген.

Мағлыўматлардың структурасы дегенде улыўма жағдайда мағлыўматлар элементлери көплиги ҳәмде олар арасындағы байланыслар көплиги түсиниледи. Бундай анықлама бериў мағлыўматларды структуралаўдың усылларын өз ишине алып, бирақ анық мәселелерди шешкенде оның қандайда бир аспектлери қолланылады. Сонлықтан мағлыўматларды ажыратыўда қосымша классификациялаў пайдаланылады.

«Мағлыўматлардың физикалык структурасы» түсиниги мағлыўматлардың компьютердиң ядында физикалық сәўлелендирилиўин көрсетип, сақлаў структурасы, ишки структура ямаса ядтың структурасы депте аталады.

Компьютер ядында сәулелениўди есапқа алмаған ҳалда мағлыўматлар структурасын қарастырыў абстракт ямаса логикалық структура деп аталады. Сонлықтан логик структураларды физик структураларға ҳәм керисинше өткериўши процедуралар бар болады.

Жәнеде мағлыўматлардың структурасын әпиўайы ҳәм интегралласкан депте екиге бөледи. Әпиўайы мағлыўматлар структурасы дегенде қураўшы бөleklerge бөлинбейтуғын, битен асып кетпейтуғын мағлыўматлар структурасы түсиниледи. Интегралласқан мағлыўматлар структурасы бул қураўшылары әпиўайы ямаса өз гезегинде жәнe интегралласқан мағлыўматлар структурасы болады.

Мағлыўматлардың интегралласқан структуралары программалыўшы тәрепинен программаластырыў тиллери жәрдеминде мағлыўматларды интеграциялаў арқалы қуралып жаратылады.

Элементлер арасындағы байланыслардың бар ямаса жоғына байланыслы мағлыўматларды байланыспаған структурада (векторлар, массивлер, қатарлар, стеклер, нәўбетлер) ҳәмде байланысқан структуралар (байланысқан дизимлер) депте ажыратыўға болады.

Мағлыўматлар структураларының әҳмийетли белгилериниң бири бул оның өзгериўшеңлиги (структура элеменлериниң санының ямаса олар арасындағы байланыстың өзгериўи) болып табылады. Бунда мағлыўматлар элементлериниң мәнислериниң өзгериўи есапқа алынбайды. Мағлыўматлар структурасы өзгериўшеңлигине байланыслы базалық, статистикалық, ярым стастистикалық, динамикалық ҳәмде файллық структураларға ажыратылады.

Базалық, статистикалық, ярым стастистикалық ҳәм динамикалық струтурадағы мағлыўматлар оператив яд ушын характерли болып, көбинесе оператив структуралар деп аталады. Файллық структуралар бул сыртқы яд ушын мөлшерленген болады.

Мениң питкериў қәнийгелик жумысымызда баслы мақсет оператив яд ушын сызықлы емес структурадағы мағлыўматлардың байланысларын

сыртқы ядта (файлларда) сәўлелендириў ҳәм программаластырыўда қолайлы алгоритмлер менен қолланыў усылларын жаратыў болып табылады.

Енди биз алға қойған мақсетимизде қолланылатуғын мағлыўматлардың структураланған түрлери менен танысып өтейик:

Вектор (бир өлшемли массив) – фиксерленген сандағы ,бир түрдеги элеменлерден турыўшы мағлыўматлар структурасы.

Мысалы: (1; 2; 15; 2; 22) ямаса (Артықбаев; Даўлетов; Халиков)

Массив – базалық деп аталыўшы, бир түрдеги элементлер избе излиги.

Мысалы: $\begin{pmatrix} 1 & 5 & 4 \\ 2 & 1 & 15 \\ 3 & 11 & 5 \end{pmatrix}$ ямаса $\begin{pmatrix} Xalikov; & Babanazar \\ Artiqbaev; & Iqlasbay \end{pmatrix}$

Көплик – бул бир түрдеги жәнеде қайталанбайтуғын мәнистеги мағлыўматлар структурасы.

Буған мысал ретинде алып текте элементлер мәнислери текте бир мәртебе ушырасатуғын формада жоқарыдағы массивлерди алыўға болады.

Жазыў – ҳәр қыйлы мағлыўматлар түрлеринен тәртиплестирилген майданлар көплиги.

Мысалы: (Risnazarov; Ansatbay; 1975; TITU; assistent)

Таблица – бирдей жазыўлардың избе-изликлеринен турыўшы структура.

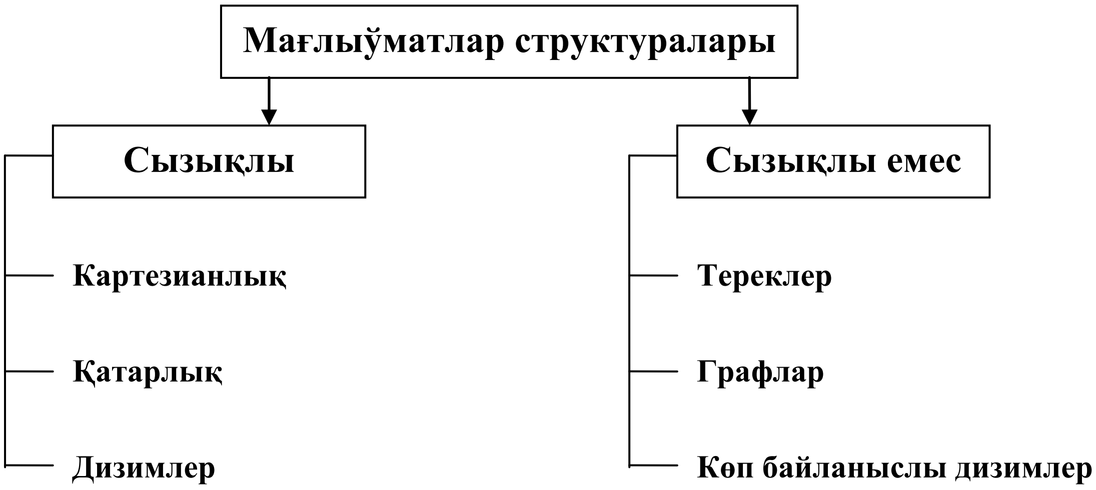
Мысалы: $\begin{pmatrix} Risnazarov; & Ansatbay; & 1975; & TITU; & assistent \\ Xalikov; & Babanazar; & 1990; & TITU; & student \end{pmatrix}$

Дизимлер - элементлерди қосыў ҳәм алыў операцияларын қолланыў мүмкин болған, өзгериўши сандағы элементлерден турыўшы тәртиплестирилген көплик.

Қоңсы болған элементлер арасында қатнасқа ийе ҳалатында тәртиплескен дизим, сызықлы дизим деп аталады.

Мағлыўматлар структурасының көзге көринерли белгиси бул оның элементлери арасындағы байланыслардың тәртиби болып табылады.

Сонлықтан структуралардың белгилерине қарап оларды сызықлы-тәртиплестирилген (сызықлы) ҳәмде сызықлы емес-тәртиплестирилген (сызықлы емес) структуралар деп ажыратылады.



Компьютер ядында өз-ара байланыста жайласыў харектерине байланыслы сызықлы структураларды ядта элементлери избе-из бөлистирилетуғын структуралар (векторлар, қатарлар, массивлер, стеклер, нәўбетлер) ҳәм де ядта қәлегенше байланыстабөлистирилетуғын структураларға (бир байланыслы, еки байланыслы ҳ.т.б. дизимлер) ажыратыўға болады.

Сызықлы мағлыўматлар структурасы – бул элементлер арасындағы байланыслар қандайда бир шәртлерди орынлаў керек болмаған структуралар.

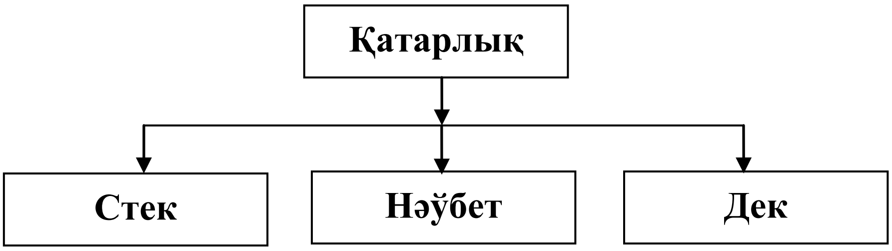
Сызықлы структуралар картезианлық, қатарлық ҳәм дизимли болып бөлинеди.

Картезианлық ямаса туўры мүйешли структуралар бул туўрымүйеш таблицалар усылында жазыў мүмкиншилигин беретуғын мағлыўматларға айтылады. Мысалы:

B = (9, 3, 6, 5) – вектор;

Z = {7, 6, 0, 2, 3} –элементлер көплиги;

Қатарлық структуралар – бир өлшемли, динамик өзгериўшеӊ мағлыўматлар структурасы болып, оған элемент қосыў ҳәм алыў мүмкиншилиги болыўы керек. Өз нәўбетинде қатарлық структураларды стек, дек, нәўбет түрлерине бөлиўге болады.

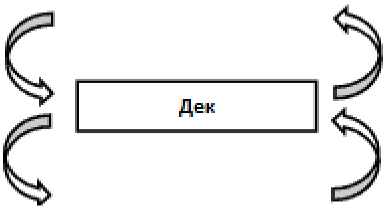


Стек – бул элементти қосыў ҳәм алыў бир тәрептен болатуғын избе-излик түриндеги мағлыўматлар структурасы. Бул жерде «ақырғы келеген – биринши кетеди» (LIFO: Last Input – First Output) принципи қолланылады. Стекке мысал етип вагонларды тәртиплестириўде қолланылатуғын темир жол разъездин алыўға болады.

Нәўбет – бул элементти қосыў бир тәрепинен, ал алыў екинши тәрепинен болатуғын избе-излик.

Бул структура FIFO (биринши келгенге – биринши хызмет көрсетиледи) принципинде ис жүритеди.

Дек – элементлерди қосыў ҳәм алыў еки тәреплеме әмелге асырыўға болатуғын избе-излик.



Барлық дизимлик структураларда мағлыўматлардың логикалық тәртиби көрсеткишлер арқалы аңлатылады. Олар ҳәр бири элементтиң мәнислерин ҳәм өзинен кейинги (алдынғы) элементке көрсеткиш берилетугын еки майданнан турады.

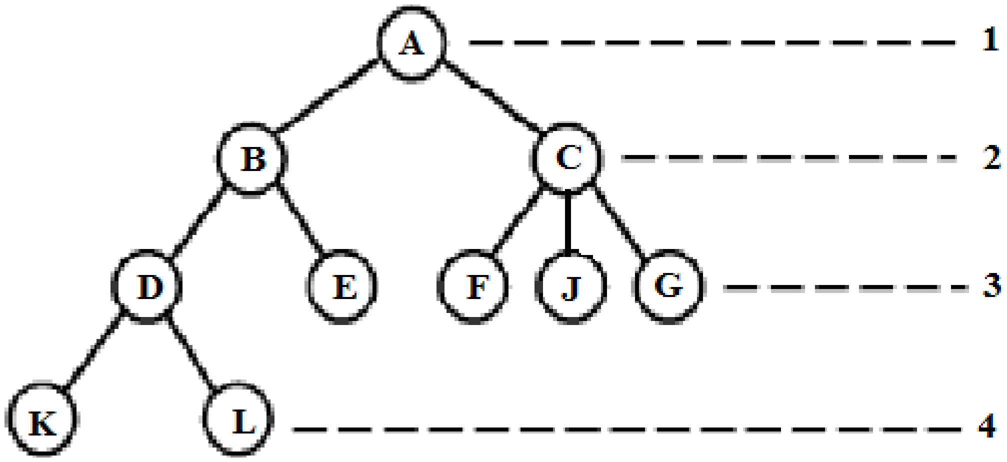
Бул жоқарыдағы қарастырып өткен мағлыўматлар структуралары элементлер избе-излиги белгили тәртипте жайластырылған ямаса анық орны көрсетилген структурар болып олар программаластырыўда кеңнен қолланылады. Булардан басқа терек, граф ҳәм көп байланыслы дизимлер формасындағы структуралар программаластырыўда жийе ушырасып олар сызықлы структуралардың жәрдеми менен қурастырылады.

Питкериў қәнийгелик жумыстың келеси бөлиминде сызықлы емес структуралар ҳәм оларды программаластырыўда қолланыў усылларын қарастырып өтемиз.

§2. Сызықлы емес структуралы мағлыўматлар ҳәм оларды программаластырыўда қолланыў

Сызықлы емес структурадағы мағлыўматлар – бул элементлер арасындағы байланыслар анық шәртлерге багынышлы болған мағлыўматлар структуралары болып, оған тереклер, графлар ҳәм көп байланыслы дизимлерди мысал етип көрсетиўге болады.

Терек көринисиндеги структуралар ҳәр-бири анық информацияны сақлаўшы ҳәм өзинен төмендеги (жоқарғы) дәрежедеги төбеге бағытланған төбелер ҳәм қабырғалардан турыўшы иерархиялық структура болып табылады. Терек – бул түйинлер (олардың биреўи түбир болады) ҳәм олар арасында иерархикалық структураны жаратыўшы қатнаслардан турыўшы элементлердиң жыйындысы болып саналады.

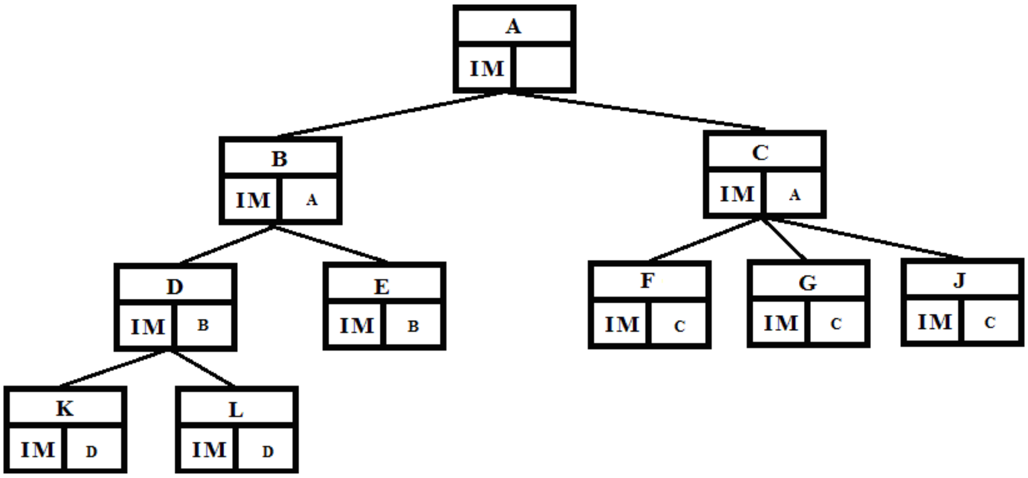


Нулинши дәрежеде жайласқан төбе теректиң түбири деп аталады (дәреже нумерациясын 1 ден басласада болады). Түбирге бирде-бир қабырға кирмейди. Өзинен бирде-бир қабырға шықпайтуғын төбелер жапырақлар деп аталады. Жоқарыдағы мысалда E, F, J, G, K ҳәм L төбелер жапырақлар болып саналады.

Бундай байланыста болған структураларды бизге белгили түрлер менен аңлатыў керек болады. Терек структурасындағы ҳәр-бир түйин өзиниң

информациялық майданы ҳәмде көрсеткиш майданына ийе болыўы керек. Көрсеткиш майданында структураның байланысын жаратыў усылына қарап өзиниң бағынатуғын ямаса оған бағынатуғын түйинлер ҳаққында жазыў болыўы тийис.

Жоқарыдағы мысалда көрсетилген теректи информацион майданлары ҳәм көрсеткиш майданларға ийе жазыўлар арқалы өзи тийисли түйинди таныйтуғын усылында структураласақ төмендегише болыўы керек.

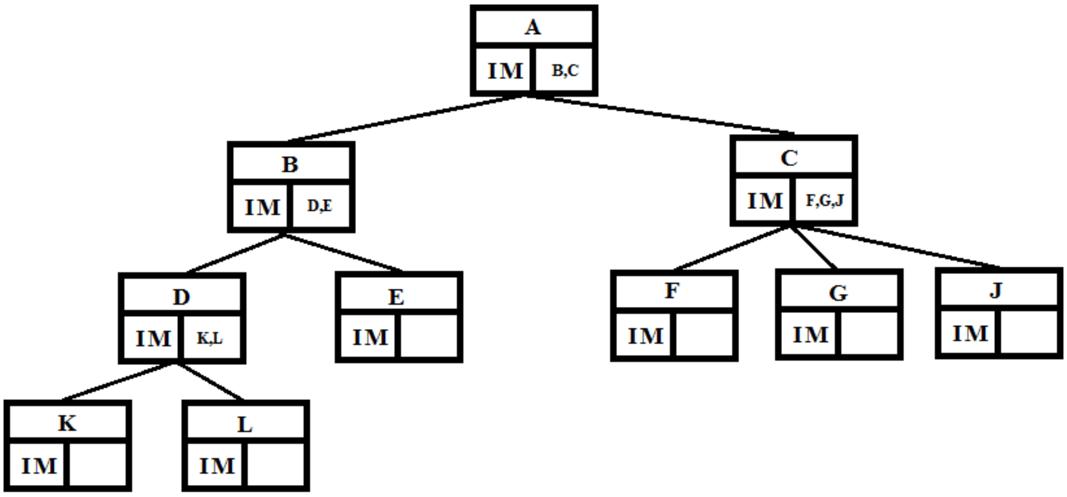


Бунда түбир түйиннен басқа барлық түйинлердиң көрсеткиш майданларында жазыў бар болады. Ҳәмде бул жазыўларда өзи тийисли болған түйинлер (ата түйин) жазылады. Бирақ өзине тийисли болған түйинлер ҳаққында мағлыўматқа ийе болмайды. Программаластырыўда терек көринисиндеги мағлыўматлар структураларын қолланыўда массивлер ҳәм жазыўлардан пайдаланылады. Көрсеткишлерди жоқарыдағы усылда алыў ҳәр-бир түйин ушын адрес, информация майданы, көрсеткиш майданы, түйинниң дәрежесин көрсетиўши майдан ҳәмде бир атаға тийисли түйинлердиң избе-излигин көрсетиўши майданлар жәмленген жазыў (улыўма майданлары бесеў болған) арқалы аңлатыўға болады.

Биз терек түриндеги мағлыўматлар структурасын бул усылда жаратсақ онда қәлеген бир элемент ҳаққында толық мағлыўматқа ийе болыў ушын оның жазыўларынан басқа бул түйинге тийисли түйинлер барма, бар болса

олар қайсылар деген сораўға жуўап алыў ушын бул элемент турған дәрежеден кейинги дәрежедеги түйинлердиң көрсеткиш майданларын анализлеп шығыўымыз керек.

Енди биз усы теректи төмендегише усылда қабыл етейик.



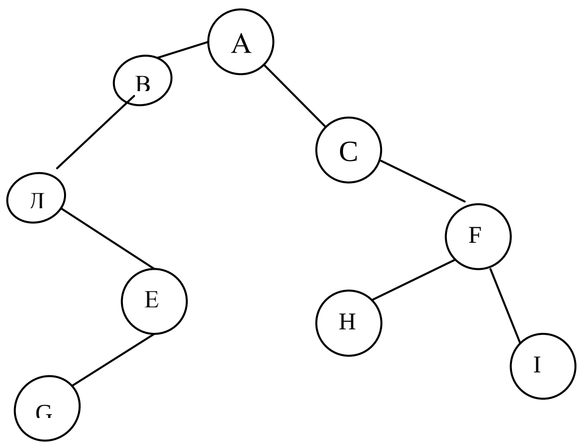
Бунда көрсеткиш майданларына усы түйинге тийисли болған (егер бар болса) түйинлер жазылады. Жапырақ түйинлердиң көрсеткиш майданларыбос ҳалында болады. Бунда түйинниң ҳалатын толық тәрийплеў ушын адрес, информация майданы, көрсеткиш майданы ҳамде түйинниң дәрежесин көрсетиўши майданлар жәмленген жазыў арқалы аңлатыўға болады. Жоқарыдағы усылға қарағанда бир майданды кемейтириўге ерискенимиз бенен, түйинге тийисли пәски дәрежедеги түйинлер текте жалғыз ямаса анық белгили санда болса бир майданның ишине жазыў ҳәмде оны программа ишинде қолланыў жеңиллеў болыўы мүмкин. Кери жағдайда программада қолланыў ушын қолайлы болыў ушын жазыўға өз алдына майданлар қосыўға туўры келеди. Ямаса бир майданның ишине арнаўлы қабыллаған белгилер менен ажыратып жазыўға болады. Бундай усылда жазыў программада мағлыўматларды ажыратыўда қосымша әмеллерди орынлаўға алып келеди. Жәнеде түйинниң ата түйинин табыў ушын өзинен бир бағана жоқары түйинлерди қарастырып шығыўымыз керек.

Бул усылды терек усылындағы структуралардың бинар терек түринде қолланыўы жеңил болыўы мүмкин.

Бинар терек деп бос болған ямаса түбир деп аталыўшы биринши элементке ийе болып, қалған элементлериниң хәр бири кесилиспейтуғын еки үлес көпликлерге бөлинген шекли элементлер көплигине айтамыз. Бул үлес көпликлер дәслепки теректиң шеп хәм оң үлес көпликлери делинеди.

Бинар теректиң хәр бир элементи терек түйини делинеди.

Төменде бинар теректи сүўретлеў келтирилген.



Қатаң бинар терек

Терек 9 түйиннен ибарат. А-терек түбири оның шеп үлес тереги В-түбирге, ал оңы С түбирге ийе. Бул А- дан шығыўшы еки шақа менен шептеги В ға оңдағы С- ға барады. В оның шеп ямаса оң үлес тереги болсын, онда А – В- ның жолы, В шеп ямаса оң баласы делинеди. Балаларға ийе болмаған түйин жапырақ (лист) деп аталады. n1-түйин n2- түйинниң атасы деп аталады. Егер n1 n2 ниң әкеси ямаса әкесиниң әкеси болса. Мәселен А –G ның атасы, H-C ниң урпағы. n2- түйин n1 түйинниң шеп урпағы делинеди егер n2 n1 диң шеп баласы ямаса n1 шеп баласының урпағынан болса еки түйин ағалы инили делинеди егер олар бир әкениң баласы болса. Егер бинар теректиң хәр бир жапырақ болмаған түйини бос емес оң хәм шеп үлес терекке ийе болса онда қатаң бинар терек делинеди. Қатаң бинар терек n жапыраққа ийе болса онда хәмме ўақытта 2n-1 түйинге ийе болады. Бинар

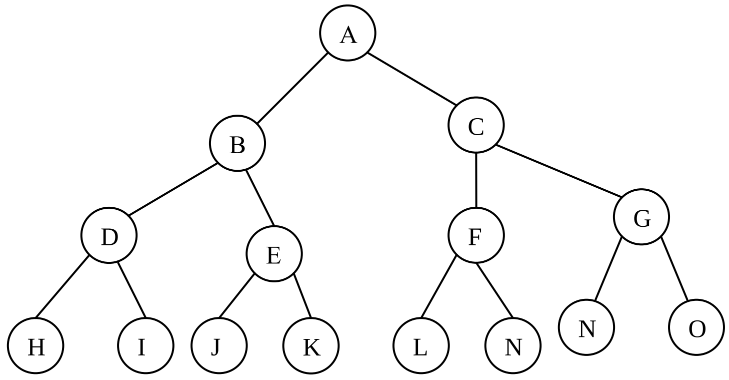
теректе түйинлер дәрежеси былай анықланады. Терек түбири 0 ши дәрежеде, басқа қәлеген түйин дәрийжеси өзиниң әкесиниң 1 дәрежесинен артық болады. E – 2 дәрет, H – 3 дәрет түйинлер.

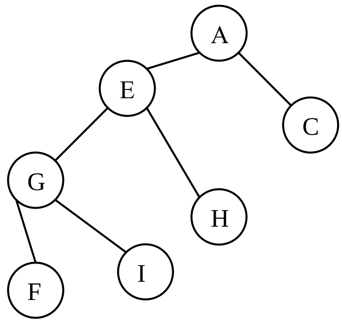
Бинар теректиң тереңлиги деп оның жапрақларының максимал дәрежесине айтамыз. n - дәрежедеги толық бинар терек – бул сондай терек болып ондағы n нен киши дәрежедеги ҳәр бир түйин жапырақ болып, n нен киши дәрийжедеги ҳәр бир түйин бос емес шеп ҳәм оң үлес терекке ийе болады.

Дерлик толық бинар терек деп сондай бинар терекке айтамыз, сондай терис емес пүтин K саны бар болып:

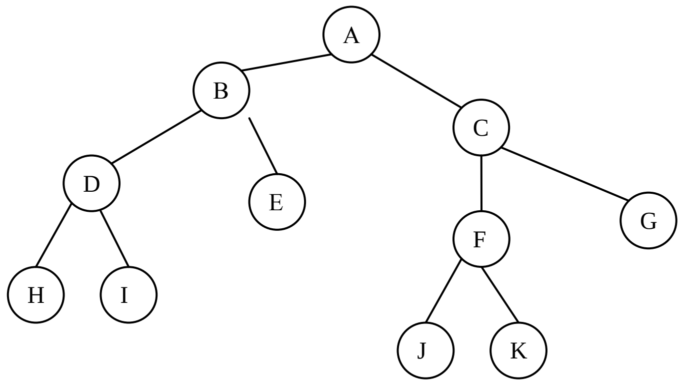
Теректиң ҳәр бир жапрағы K ямаса K+1 дәрежеге ийе

Егер түйин K+1 дәрежедеги оң урпаққа ийе болса онда оның барлық шеп урпақлары жапырақ болып K+1 дәрежеге ийе болады.

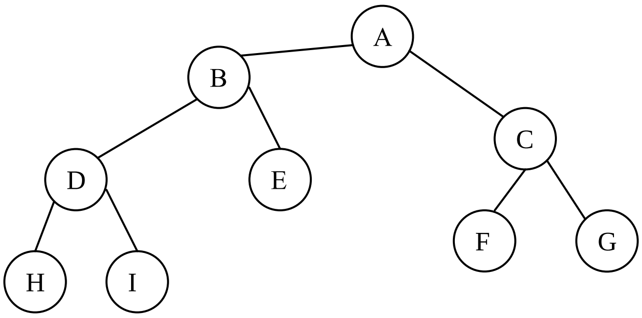




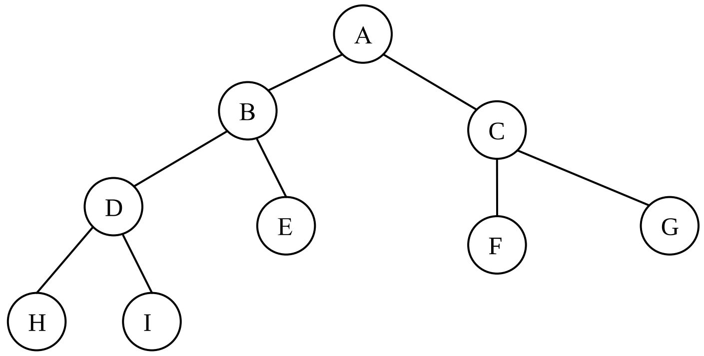
а) Қатаң бинар терек дерлик толық емес бинар терек 1 шәрт бузылған



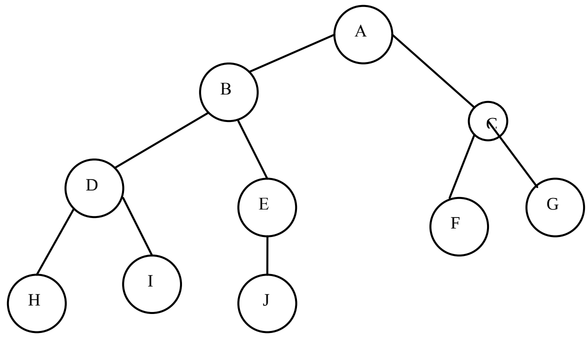
б)   Қатаң бинар терек дерлик толық емес (2- шәрт бузылды)



в) Қатаң бинар терек дерлик толық бинар терек



д)



г) Дерлик толық бинар терек қатаң емес

лев сын= узв отец

прав сын = узв отец+1

Дерлик толық қатаң бинар терек n – жапырақлы 2n-1 түйинге ийе болады.

Дерлик толық бинар терек қатаң емес n жапырақлы 2n- түйинге ийе болады. n - түйинге дерлик толық бинар терек қатаң бинар терек болады егер n- тақ болса.

Бинар терек ушын элементар әмеллер орынлы. Егер nd түйинге ийе бинар теректиң көрсеткиши p болса, онда info(p) функциясы nd түйинниң ишиндеги мағлыўматты билдиреди. left(p), right(p), father(p) ҳәм brother(p) функциялары сәйкес түрде nd түйинниң шеп баласы, оң баласы, атасы ҳәм инисиниң көрсеткишлерин аңлатады. Бул функциялар бос көрсеткишти анықлайды, егер nd түйин шеп балаға, оң балаға, атасы ҳәм инисине ийе болмаса.

Логикалық функциялар isleft(p) ҳәм isright(p) true мәниске ийе болады егер nd түйин базы бир түйинниң сәйкес түрде шеп ямаса оң баласы болса егер ондай болмаса онда ол функцияллар false (жалған) мәнисин қабыл етеди.

isleft(p), isright(p), ҳәм brother(p) функцияларын left(p), right(p), father(p) функциялары жәрдеминде әмелге асырыўға болады.

isleft(p) функциясы алгоритми

```
function  isleft(p)
q=father(p)
if q=null then isleft= false
else if left(q)=p then isleft= true
else isleft false
end if
```

```
end if
return
```

Усыған уқсас isright функциясын жазыў мүмкин.

brother(p) функциясы алгоритими

```
function  brother(p)
 if father(p)=null
 then brother(p)=null
else if left(p)
 then brother = right (father(p))
else brother = left (father(p))
end if
end if
return
```

Бинар тереклерди дүзиўде make tree, set left, set right әмеллери пайдалы болады.

make tree(x) функциясы информациялық майданда x болған бир түйинли бинар терек дүзеди ҳәм оның түйини көрсеткишин анықлайды. set left(p,x) функциясы информациялық майданда x болған шеп баласы болмаған nd түйинге ийе бинар Терек ушын оның шеп баласын дүзеди.

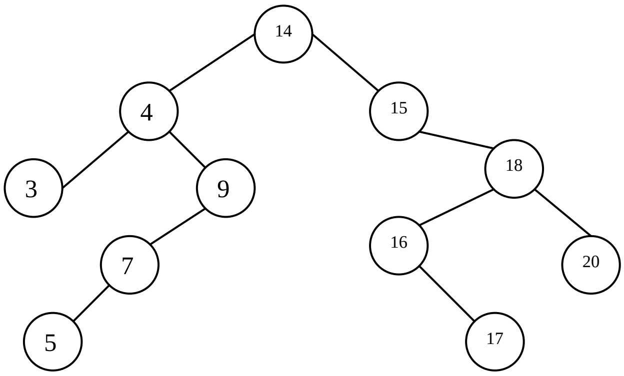
set right(p,x) функциясы p көрсеткиш бинар теректиң оң баласын дүзеди. Бинар теректиң қолланылыўына мысал көремиз.

Бинар терек еки жағдайдан биреўин таңлап алыўда жақсы жәрдем береди.

Егер бизге санлар дизими берилген болып олар ишинен барлық дубликатларды анықлаў керек болсын. Буны шешиў ушын санлардың ҳәр бирин оннан алдынғысы менен салыстырып табыўғада болады, бирақ бунда көп салыстырыў керек болады. Егер бинар терек қолលансақ санын азайтыў мүмкин.

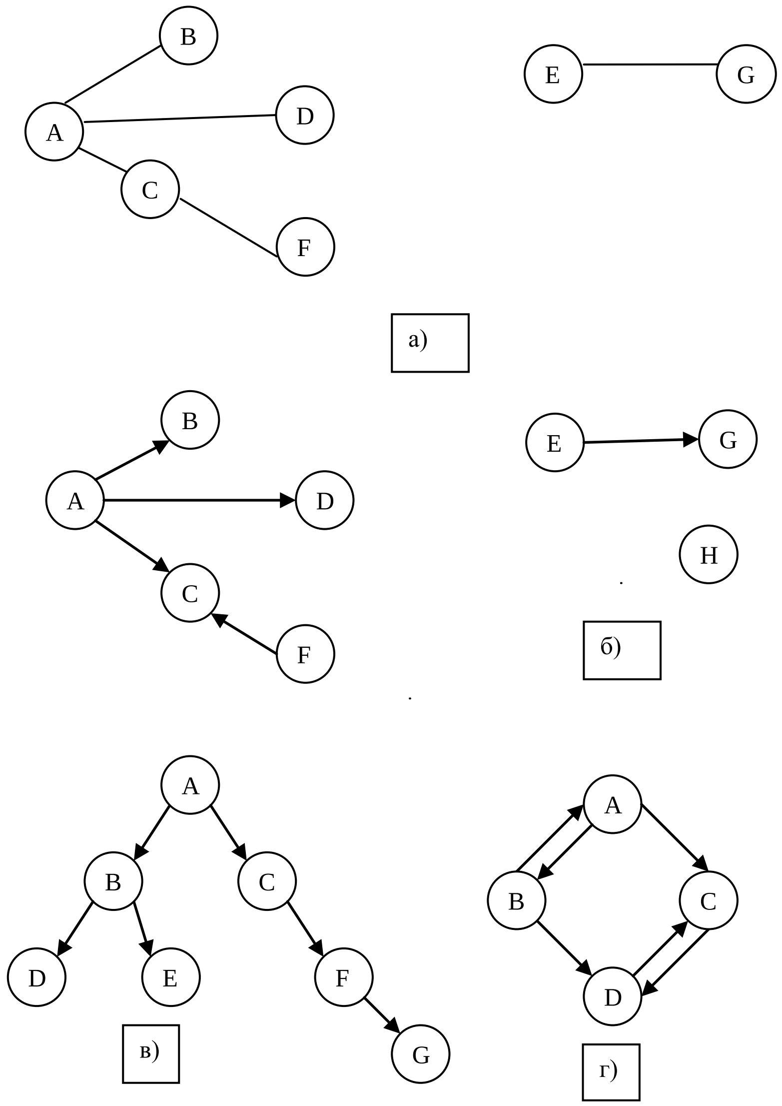
Шешилиўи: биринши санды алып оны түйинге жайластырамыз ҳәм оны Терек түбири деп есаплаймыз. Оннан кейинги ҳәр бир сан түбирдеги сан

менен салыстырылып егер ол сан тең болса дубликат деп есапланады егер ол сан түбирдеги саннан киши болса онда түбирдиң шеп баласына, ал үлкен болса оң баласына жайластырылады. Одан кейинги сан түбирден киши болып ҳәм оның шеп баласынан да киши болса онда шеп баласының шеп баласына, үлкен болса шеп баласының оң баласына жайластырылады ҳәм процесс даўам етеди,



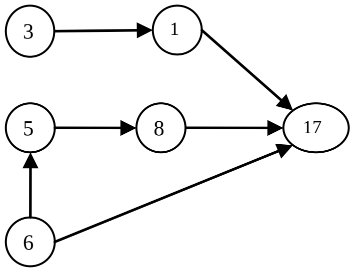
көринисине келтириледи.

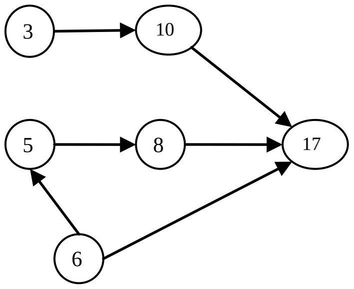
Графлар төбелерден ҳәм доғалар көплигинен турады. Графта ҳәрбир доға түйинлер жубы менен көрсетиледи. 13.1-а сүўретте граф көрсетилген оның түйинлер көплиги {A,B,C,D,E,F,G,H}, доғалар көплиги {(A,B),(A,D),(A,C),(C,D),(C,F),(E,G),(A,A)}. Доға дүзиўши түйинлер жубы тәртиплескен болса, онда граф бағытланған граф делинеди.

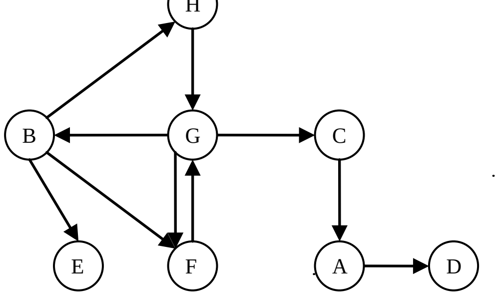


Сүўретлерде 3 бағытланған графлар көрсетилген. Түйинлер арасындағы аралықлар доға дүзеди. Стрелканың басы түйинниң екиншисин көрсетеди, ал оның кейини биринши түйинди көрсетеди. б сүўреттеги графтың доғалар көплиги {<A,B>, <A,C>, <A,D>, <C,D>, <F,C>, <E,G>, <A,A>} бунда бағытланған графлар доғалары мүйеш қаўыслар менен көрсетилген.Соны айтып өтиў керек, граф терек болыўы шәрт емес, бирақ терек шәртли түрде граф болады. Теректеги (в сүўрет) атадан балаға көрсеткиш граф доғасы ролин атқарады. Түйинниң доғасы болыўы шәрт емес. N- түйин x –доғаға инцидентли делинеди, егер N x- доғаны дүзиўши тәртиплескен еки түйинниң биреўи болса. Түйинниң дәрежеси деп түйинге инцидент доғалар санына айтамыз. N –түйинге кириўдиң ярым дәрежеси деп N -түйинге тирелген стрелкалар санына айтамыз. N- түйиннен шығыўдың ярым

дәрежеси деп оннан шыққан стрелкалар санына айтамыз. Мысал ушын г сүўретте А түйинниң ярым дәрежели кириўи -1 ге тең, ярым дәрежели шығысы -2 тең. N- түйин М- түйин менен қоңсы делинеди, егер М менен N ге доға бар болса. А–көплигинде берилген R қатнас – бул А элементлериниң тәртиплескен элементлер жубы. Егер <x,y> жубы R қатнасы ағзасы болса, онда x y пенен R қатнаста болады деп айтамыз.Мәселен А= {3,5,6,8,10,17 } көплиги болса , онда {<3,10>, <5,6>, <5,8>, <6,17>, <8,17>, <10,17>} көплиги А дағы қатнас деп аталады.Бул қатнас былай сүўретлениўи мүмкин : x ҳәм y берилген қатнаста болады, егер x ҳәм y А ға тийисли болып , x y тен киши ҳәм y тиң x қа бөлингендеги қалдық тақ сан болса. Элемент <8,17> айтылған қатнас элементи болады, себеби 8 17 ден киши ҳәм 17 ни 8 ге бөлсек қалдық 1 ге тең . Қатнас граф түринде берилиўи мүмкин , оның түйинлери тийкарғы көпликти, доғалары тәртиплескен қатнаслар жубын анықласа. 13.2-а сүўретте жоқарыдағы сүўретленген қатнасты көрсетиўши граф берилген.





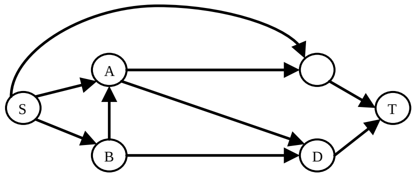


Графтың ҳәр бир доғасы менен бир мәнис байланған болыўы мүмкин б сүўрет. Бул жерде ҳәр бир доғаға стрелка менен көрсетилген түйинлердеги санларды бирине бирн бөлиўден шыққан қалдықты байланыстырамыз. Бундай доғасы менен қандай да бир мәнис байланысқан графқа өлшемли граф ямаса тармақ деп атайды. Ҳәр бир доға менен байланысқан мәниске оның салмағы делинеди.

Енди графлар менен жумыс ислеўде пайдалы болған әмеллер менен танысайық. Join(a,b) әмели графтың а түйининен b түйинине доға қосады, егер доғасы болмаса. joinwt(a,b,x) әмели өлшемли граф ушын а түйиннен b түйинге доға қосады ҳәм оның салмағын x деп алады. remv(a,b) ҳәм remvwt(a,b,x) әмеллер сәйкес граф ҳәм өлшемли графтан а дан b ға шекемги доғаны алып таслайды. Adjacent(a,b) функциясы b түйини а түйини менен қоңсы болса онда «рас» мәнисин қабыл етеди, егер керисинше болса «жалған» мәнисин қабыл етеди. а түйиннен b түйинге шекем K-узынлықтағы жол k+1 сандағы сондай n1, n2, …nk+1 түйинлер избе-излиги менен анықланады олар ушын n1=a nk+1=b ҳәм adjacent(ni,ni+1) функциясы 1 k шекемги аралықтағы барлық i лер ушын рас мәнисин қабыл етеди. Егер қандай да бир r ушын а түйиннен b түйинге шекем к узынлықтағы жол бар болса онда а түйиннен b түйинге жол бар деп есапланады. Түйиннен оның өзине шекемги жолды цикл деп атайды. Егер графта цикл бар болса оны циклы деп атайды, болмаса ацикллы деп атайды. 3-сүўреттеги графты қараймыз, онда А түйиннен С түйинге узынлығы 1 ге тең жол бар, В түйиннен G түйинге узынлығы 2 тең 2 жол бар, А дан F ке бир жол узынлығы 3 ке тең, В дан В ға цикл бар, F тен F ке, H тан H қа да цикллар бар.

Ағымлар мәселеси

Биз реал әмелий мәселени қарайиыз ҳәм оны өлшемли граф жәрдеминде шешемиз. Оны ҳәр қыйлы қойылыўда қараўға болады. мейли бизге суў өткизиўши трубалар системасы берилген болсын. ҳәр бир доға трубаны билдиреди ондағы сан трубаның суў өткизиў мүмкиншилигин аңлатсын.



S тең Т ға бир бағытта суў ағады, S суў көзи, Т жыйналыў пункты болсын. Биз S тен Т ға ағып өтиўши суў ағымын максималластырыўымыз керек. С(а, в) деп а дан в ға ағып өтиўши трубаның өткизиў мүмкиншилиги функциясын белгилеймиз. С(а, в) трубаның суў өткизиў мүмкиншилиги мәнисине тең. Егер а дан в ға труба жоқ болса С(а, в)=0 болады. f(а, в) ағым функциясы болсын ҳәм ол а дан в ға ағып өтиўши суў муғдарына тең болады. f(а, в)$\leq$ С(а, в) болады. Мейли V S тен Т ға ағып келиўши суў көлеми болсын. Онда

$$\sum f(s,x) = V = \sum f(x,T) \qquad (1)$$

$$\sum f(x,y) = \sum f(y,x) \qquad \forall x \neq S,T$$

х түйинниң шығыўшы ағымы деп оннан шығып кетиўши ағымлар суммасына, кириўши ағымлар деп оған келип түсиўши ағымлар суўммасына айтамыз. Демек S тиң шығыўшы ағымы =Т ниң кириўши ағмына =V.

Демек трубаның өткизиўшилик мүмкиншиликлерине қарап суў ағымын жақсылаў мүмкин, яғный суў ағымы функциясы мәнислерин көбейтиўге болады. Оның еки жол бар, бириншиси S тең Т ға шекемги жолларды қарап шығып олар ишинде суў өткериў мүмкмншиликлери суў ағымынан үлкенирек болған жол табылса онда жолдың доғаларының минимал суў өткизиўшилик мәнисине шекем суў ағмын көтериўге болады.

Алгоритм

Ҳәр бир функцияның мәнисине қараймыз

S тен Т ға шекемги сондай жолды таңлаў керек Т ға келип түсиўши ағым а>0 шамаға артсын.

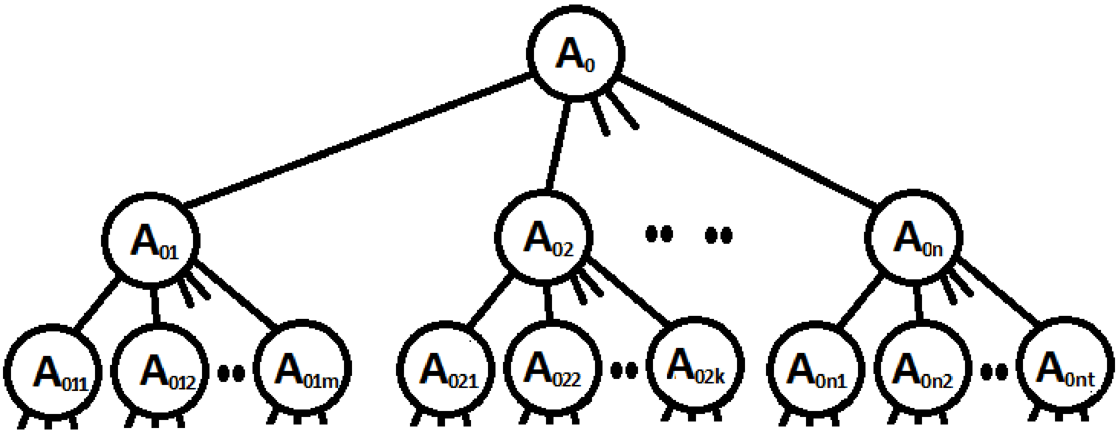
Егер бундай жол табылмаса кейинге қайтарылады.

Ҳәр бир түйинде ағымда а мәнисине көбейтиў

2 – адымға өтилсин.

§3. СЫЗЫҚЛЫ ЕМЕС СТРУКТУРАЛЫ МАҒЛЫЎМАТЛАР УШЫН МАҒЛЫЎМАТЛАР БАЗАЛАРЫН ҚОЛЛАНЫЎ.

Биз төмендеги көринистеги терек формасындағы байланысқан мағлыўматлар ушын олардың байланысын, информациялық майданлары мәнислерин жарата алатуғын мағлыўматлар базасының моделин ислеп шығайық.



Бул мағлыўматлар структурасында n,m,k ҳәм t лар пүтин санлар болып ҳәр-бир түйиннен щығатуғын шақалар санын болып, түйин туйық болса 0 ге тең болады. A_0 түбир түйинин ҳеш-бир түйинге бағынбағанлығы себепли оны таблицада көрсетип кетпесекте болады. Бирақ оған тийисли информация майданын тийисли таблицада көрсетиўимиз керек. Деген менен басқа түйинлер ҳәмде оларрдың информация майданлары усы түбир түйинге тийисли болғанлықтан, терек түриндеги мағлыўматлар структураларындағы ҳәр-бир түйин өзиниң информация майданына ийе болатуғынлығы, қосымша шақаларға ийе болыўы мүмкинлиги көринеди.

Жәнеде бундай мағлыўматлар тийкарында жаратылған информациялық системалардың ислеў процессинде қандайда ўақыт аралықларында бир бағдардағы информацияларды алып таслаў, қосымша бағытта информациялар қосыў, белгили түүйинлерди басқа түйинлерге өткериў ҳ.т.б.

керек болып қалады. Сонлықтан мағлыўматлар базаларында сәўлелендириўде булардың барлығын есапқа алыў керек.

Жоқарыда берилген схемадағы мағлыўматлар структурасындағы түйинлердиң бир-бирине байланысын төмендеги таблицада көрсетип өтемиз.

Түйин идентификация номери (ID)	Түйин аты	Түйинниң ата түйининиң ID	Түйинниң қатар номери	Түйинниң дәрежеси
1	A01	0	1	1
2	A02	0	2	1
..	..	0	3	1
n	A0n	0	n	1
n+1	A011	1	1	2
n+2	A012	1	2	2
…	…	1	3	2
n+m	A01m	1	m	2
n+m+1	A021	2	1	2
n+m+2	A022	2	2	2
…..				2
…..				2
n+m+k+t	A0nt	n	…	2

Бул таблицадағы байланыс бойынша ҳәр-бир түйин өзиниң ата түйинин таныйды бирақ өзине тийисли шақа түйинлерди билмейди. Сонлықтанда программаластырыў даўамында өзине тийисли шақа түйинлерди табыў керек жағдайларда белгили излеў, саралаў алгоритмлеринен пайдаланамыз.

Питкериў қәнийгелик жумысты орынлаўда үйренилген материалларға тийкарланып, берилген әмелий мәселе ушын информациялық системаның мағлыўматларының базасын ҳәмде интерфейсин ислеп шығайық.

Бизге кафедрамыздың питкериўшилериниң питкериў қәнийгелик жумысларына мәмлекетлик аттестация коммисиясының жумысларын автоматластырыўшы системаны ислеп шығыў тапсырылған болсын.

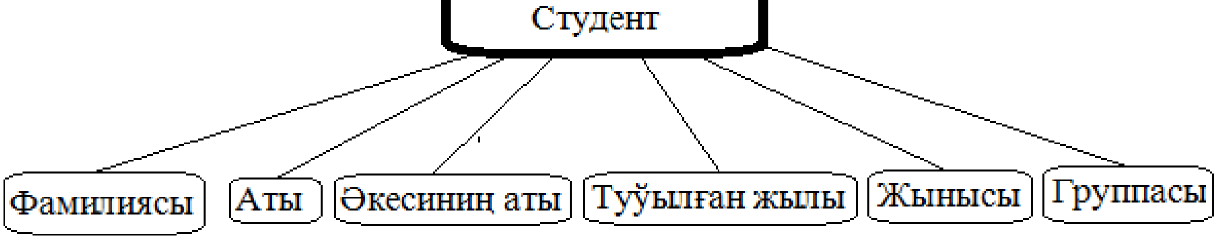
Кафедраны усы жылы еки қәнийгелик бойынша 73 студент питкерип шығыў режелестирилген болып, оларға қәнийгелик жумыслары темалары, илимий басшылары, илимий мәслахатшылар ҳәмде қорғаў сәнеси буйрық пенен бекитилген. Жәнеде мәмлекетлик аттестация комиссиясы ағзалары бекитилген. Деканат тәрепинен ҳәр-бир студенттиң оқыў даўамында пәнлерди өзлестириў көрсеткишлери жыйындысы берилген болады.

Мәмлекетлик аттестация комиссиясы ҳәр-бир студентти ажыратылған ўәкытта тыңлап, оған қосымша сораўлар берип, кейнинен студентти егер питкериў жумысын қорғады деп санаса 100 баллық системада бал қойыўы керек болады. Комиссияның ҳәр-бир студент бойынша усы ис ҳәрекетиниң ҳәммеси протоколластырылып қағаз түрринде шығарылыўы керек болады.

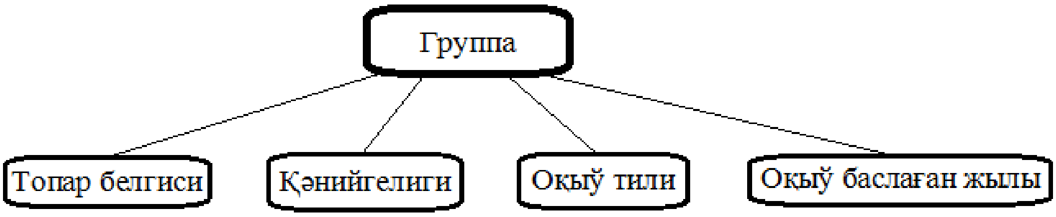
Бизиң проектлестирип атырған системамыз Веб-сервер ушын мөлшерленип, қәнийгелик қорғаў жумысы болып атырған кабинетте интернет болғанда ямаса бир компютерде локал сервер жәрдеминде жумыс ислеўи мүмкин.

Система тийкарынан аттестация комиссиясының хаткериниң жумысын жеңиллетип, жәнеде комиссия жумысы бойынша есабатларды тез таярлаў ушын қолланылады. Қағаз жумысларын азайтады.

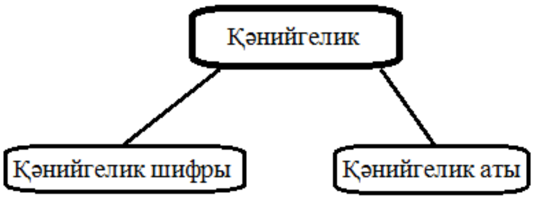
Системаның жумыс ислеў мақсетин анықлап алыўымыздан кейин енди керекли мағлыўматларды анализлеп шығайық.



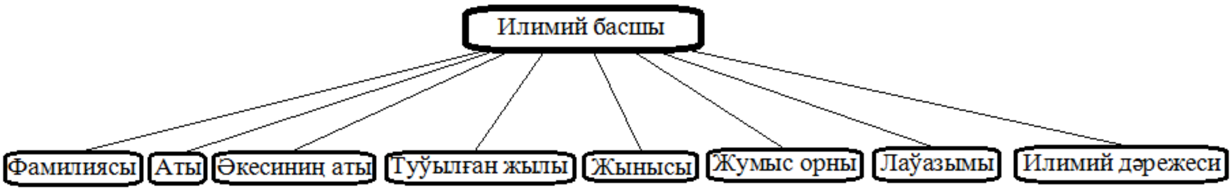
Бунда студентке тийисли мағлыўматлар берилген болып «группасы» на группалар ҳаққындағы мағлыўматлардың ишинен тийисли группа қойылады.



Группалар бир-биринен ажыратылатуғын тийкарғы мағлыўматларды минимал түрде алғанда бул жерде «Қәнийгелиги» не мәнис қәнийгеликлер хәқыында мағлыўматлардан алыныўы тийис.



Бул үш топардағы мағлыўматлар текте студентти аңлатыў ушын ислетиледи. Енди илимий басшы хәмде илимий мәслаҳатшыларды аңлатыўшы мағлыўматларды қарастырып өтейик.



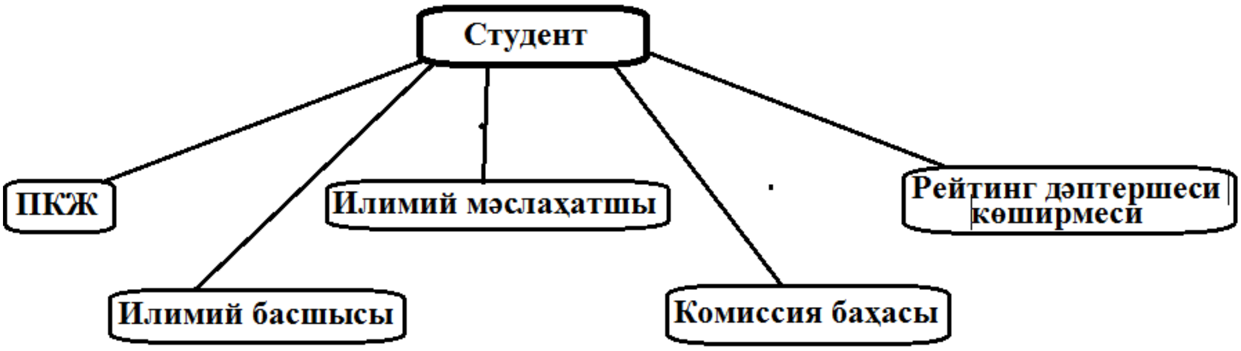
Илимий мәслаҳатшылар туўрысында да тап сондай мағлыўматлар қайталанады. Сонлыктан еки түрдеги мағлыўмат ушын бир таблицадан пайдаланамыз.

Бул мағлыўматлардан басқада керекли мағлыўматлардан: питкериў қәнийгелик жумысының темалары хаққындағы мағлыўмат, студенттиң төрт жыл даўамында оқыған пәнлери, олардың саат көлемлери хәм хәр-бир студенттиң хәр-бир пәннен топлаған баллары туўрысындағы мағлыўматлар есапқа алыныўы тийис.

Системаның жетерли информацияларды бериўде жоқарыдағы мағлыўматлардан пайдаланыўы менен бир қатарда қорғаў ўақтында студенке берилген қосымша сораўлар ҳәмде коммиссияның шешими менен студенке қойылған балл киритилиўи тийис.

Системаның мағлыўматлар базасын дүзген ўақытта қорғаў графигинде есапқа алыў зәрүр болады.

Бул мағлыўматлар барлығы студен әтирапында жәмлестирилиўи (студентке байланыстырылыўы) керек.



Бул жерде биз мағлыўматлардың тийкарғы буўынларын көрсетип кеттик. Ҳәр-бир түйин мағлыўматлары толық болыўы ушын жоқарыдағы келтирилген терек түриндеги мағлыўматлардан қуралған болады. Бул жоқарыда терек көринисинде ислеп атырған системаның мағлыўматларын ҳәмме студентлер ушын схема көринисине келтириў жәнеде бириктириў қыйын емес, бирақ программа кодының ишине жайластырыў арасындағы қатнасларды (байланысларды) бериў жүдә аўыр ҳәм система тек бир жылға ислейтуғын болып қалады. Жәнеде бул жерде егерде системаны бир неше тиллерде пайдаланыў (бир ўақыттың өзинде) мүмкиншилиги қосылыўын есапқа алсақ, программаны мағлыўматлары ишинде берилип кететуғын түрде жазыў мийнетти ақламайды.

Сонлықтан биз жоқарыда берилген мағлыўматларымыз сызықлы емес байланыста болғанлығы ушын олар ушын мағлыўматлар базаларын пайдаланыў, программаның ықшам көринисине келтирип қоймай оның өмир циклнда узайтады. Программа кодын PHP тилинде жазсақ ҳәмде

мағлыўматлар базасы ушын MySQL тилин таңласақ, онда жаратылған мағлыўматлар базасынан алыўға болатуғын мүмкиншиликтеги есабатларды жаратыў қыйыншылық туўдырмайды.

Енди жоқарыдағы берилгенлер ҳәмде киритилетуғын информациялар ушын мағлыўматлар базасының таблицаларының майданларын ҳәм олар арасындағы байланысларды керекли минимал түринде дүзип шыгайық.

«qaniygelik» - таблицасы филиалдағы қәнийгеликлер дизими:

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (2)	Қәнийгелик идентификация номери
shifr	Varchar (15)	Қәнийгелик шифры
name	Varchar (150)	Қәнийгелик аты
lang	Varchar (3)	Қәнийгелик аты берилген тили

«grupра» - таблицасы филиалдағы топарлар дизими:

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (3)	Топар идентификация номери
name	Varchar (10)	Топар белгиси (мысалы «714-10»)
qaniy_id	Integer (2)	Топар қәнийгелиги номери «qaniygelik» таблицасы «id» майданынан алынады
bas_data	Data	Топардың оқыў баслаған жылы (мысалы 2010.09.01)
oqiw_tili	Varchar (3)	Топардың оқыў тили

«student» - таблицасы студент ҳаққындағы тийкарғы мағлыўматлар ушын қолланылады:

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (5)	Студенттиң идентификация номери
surname	Varchar (20)	Студенттиң фамилиясы
name	Varchar (20)	Студенттиң аты
lastname	Varchar (20)	Студенттиң әкесиниң аты
ts_data	Data	Студенттиң туўылған сәнеси

jinisi	Varchar (1)	Жынысы («E»- ямаса «H»)
gruppa_id	Integer (3)	Студенттиң топары номери «gruppa» таблицасы «id» майданынан алынады

«mekeme» таблицасында керекли мекемелер дизими киритиледи.

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (2)	Мекемениң идентификация номери
name	Varchar (200)	Мекемениң аты
lang	Varchar (3)	Мекемениң аты берилген тили

«kaf_bolim» таблицасында мекемелердеги кафедра ҳәм бөлимлер дизими киритиледи.

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (3)	Кафедра ҳәм бөлим идентификация номери
name	Varchar (200)	Кафедра ҳәм бөлим аты
lang	Varchar (3)	Кафедра ҳәм бөлим аты берилген тили

«il_dareje» таблицасында илимий дәрежелердиң атамалары ҳәм қысқартылған түрлери дизими киритиледи.

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (3)	Илимий дәреже идентификация номери
name	Varchar (200)	Илимий дәреже атамасы
name_sokr	Varchar (12)	Илимий дәреже атамасының қысқартырылған түри
lang	Varchar (3)	Илимий дәреже атамасы берилген тили

«ilm_bm» таблицасында МАК ағзалары, илимий басшылар ҳәм мәслаҳатшылар дизими киритиледи.

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (2)	Илимий басшы ямаса мәслаҳатшы идентификация номери
surname	Varchar (20)	Илимий басшы ямаса мәслаҳатшы фамилиясы
name	Varchar (20)	Илимий басшы ямаса мәслаҳатшы аты
lastname	Varchar (20)	Илимий басшы ямаса мәслаҳатшы әкесиниң аты

ts_data	Data	Илимий басшы ямаса мәслаҳатшы туўылған сәнеси
jinisi	Varchar (1)	Жынысы («Е»- ямаса «Н»)
mek_id	Integer (2)	Мекемениң номери «mekeme» таблицасы «id» майданынан алынады
kb_id	Integer (3)	Кафедра ямаса бөлим номери «kaf_bolim» таблицасы «id» майданынан алынады
lawazim	Varchar (50)	Илимий басшы ямаса мәслаҳатшының лаўазымы
il_dar_id	Integer (3)	Илимий басшы ямаса мәслаҳатшының илимий дарежеси номери «il_dareje» таблицасынан алынады
il_ataq	Varchar (20)	Илимий басшы ямаса мәслаҳатшының илимий атағы

«mak» таблицасында Мәмлекетлик аттестация комиссиясы қурамы көрсетиледи.

Майдан аты	**Мағлыўмат түри**	**Түсиндирме**
id_mak	Integer (2)	МАК ағзасы идентификация номери, «ilm_bm» таблицасы «id» майданынан алынады
lawazim	Varchar (2)	МАК та орнын көрсетиўши майдан, МАК басшысына «mb», орынбасарына «mo», ағзаларға «ma», хаткерге «mx» белгиси қойылады
npp	Integer (2)	Есабатларда МАК ағзаларының жайласыў тәртиби

Студенттиң оқыў дәўиринде пәнлер бойынша топлаған баллары ҳақындағы мағлыўматларды сәўлелендириў ушын төмендеги таблицаларды ислеп шығамыз.

«panler» таблицасында пәнлердиң атамалары дизими киритиледи.

Майдан аты	**Мағлыўмат түри**	**Түсиндирме**
id	Integer (3)	Пәнниң идентификация номери
name	Varchar (70)	Пәнниң атамасы
lang	Varchar (3)	Пәнниң атамасының берилген тили

«bloklar» таблицасында пәнлер ажыратылатуғын блоклар атамалары дизими киритиледи.

Майдан аты	**Мағлыўмат түри**	**Түсиндирме**
id	Integer (1)	Пән блогиниң идентификация номери

name	Varchar (70)	Пән блогиниң аты
lang	Varchar (3)	Пән блогиниң аты берилген тили

«oqiw_reje» таблицасында қәнийгеликлердиң оқыў режеси дүзиледи.

Майдан аты	Мағлыўмат түри	Түсиндирме
id	Integer (4)	Оқыў режедеги пәнниң идентификация номери
id_qan	Integer (2)	Қәнийгелик номери «qaniygelik» таблицасы «id» майданынан алынады
id_pan	Integer (3)	Пәнниң номери «panler» таблицасы «id» майданынан алынады
id_blok	Integer (1)	Пәнниң номери «bloklar» таблицасы «id» майданынан алынады
max_saat	Integer (4)	Оқыў режеде пәнге ажыратылған саат көлеми

«ekzamen_baha» таблицасында студенттиң қанийгелиги бойынша оқыў режесинде көрсетилген пәнлерден баллары 100 баллық системада көрсетиледи.

Майдан аты	Мағлыўмат түри	Түсиндирме
id_st	Integer (5)	Студенттиң номери «student» таблицасы «id» майданынан алынады
id_por	Integer (4)	Оқыў режесинде пәнниң номери «oqiw_reje» таблицасы «id» майданынан алынады
ball_sto	Integer (3)	Студенттиң пәннен100 баллық системада алған баллы

«pkj_jumis» таблицасында питкериўши студент, питкериў қәнийгелик жумыстың темасы, илимий басшы, илимий мәслаҳатшы, жақлаўға қойылған сәнеси туўрысында мағлыўматлар көрсетиледи.

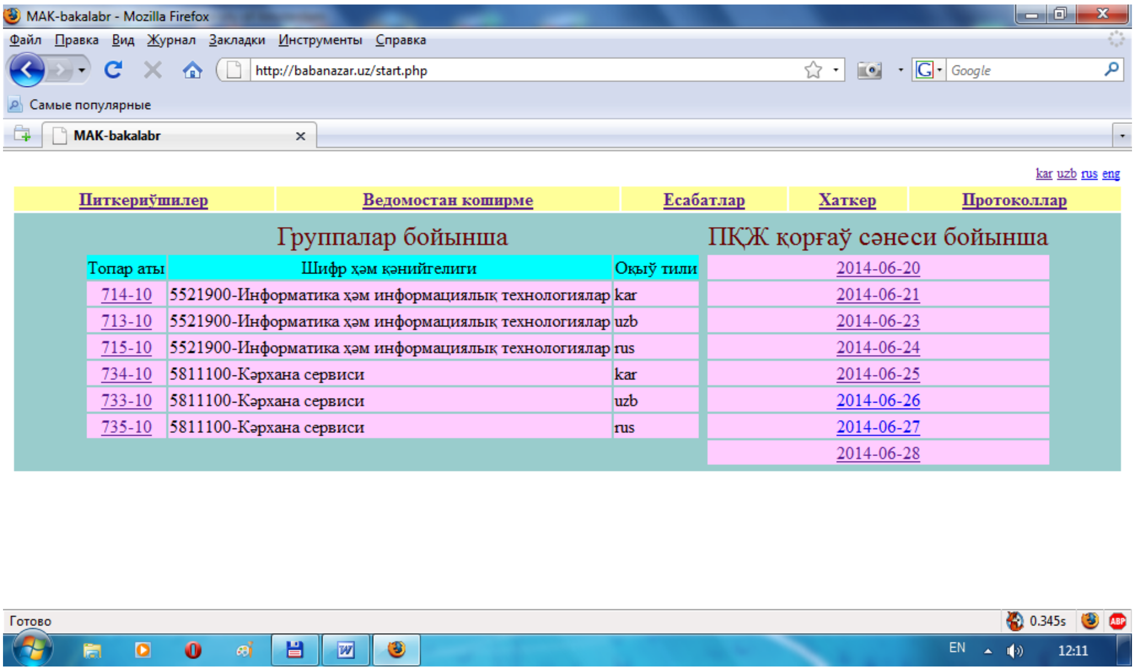
Майдан аты	Мағлыўмат түри	Түсиндирме
id_st	Integer (5)	Студенттиң номери «student» таблицасы «id» майданынан алынады
tema	Varchar (255)	ПКЖ темасы
id_ilbas	Integer (3)	Илимий басшы номери, «ilm_bm» таблицасынан «id» майданынан алынады
id_ilmas	Integer (3)	Илимий маслаҳатшы номери, «ilm_bm» таблицасынан «id» майданынан алынады
zash_data	date	Жақлаўға қойылған сәне

Жақлаў ўақтында коммисия тәрепинен берилген сораўларды, комиссияның ПКЖ ға қойған баҳасын ҳәмде студенттиң тапсырған ПКЖ сының электрон вариантының файлын көрсетиў ушын төмендеги «pkj_sorawlar» ҳәм «pkj_finish» таблицасын жаратамыз.

«pkj_soraw» таблицасы:

Майдан аты	**Мағлыўмат түри**	**Түсиндирме**
id_st	Integer (5)	Студенттиң номери «student» таблицасы «id» майданынан алынады
soraw	Varchar (255)	Сораў тексти
id_kom	Integer (3)	Сораў берген комиссия ағзасы ямаса басшысының номери «mak» таблицасының «id_mak» майданынан алынады

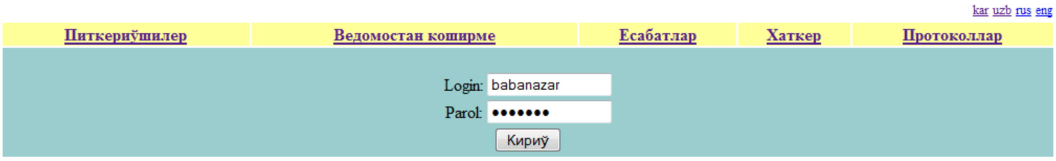
Егерде бул мағлыўматларды MySQL мағлыўматлар базасына жайластырып, Веб серверлерде жумыс ислеўши PHP тилинде программа жазып, оны веб серверге жайластырсақ онда ол бизге браузерлер арқалы тармақтан пайдалана алатуғын ахборот системасы болып, төмендеги мүмкиншиликлерди береди.



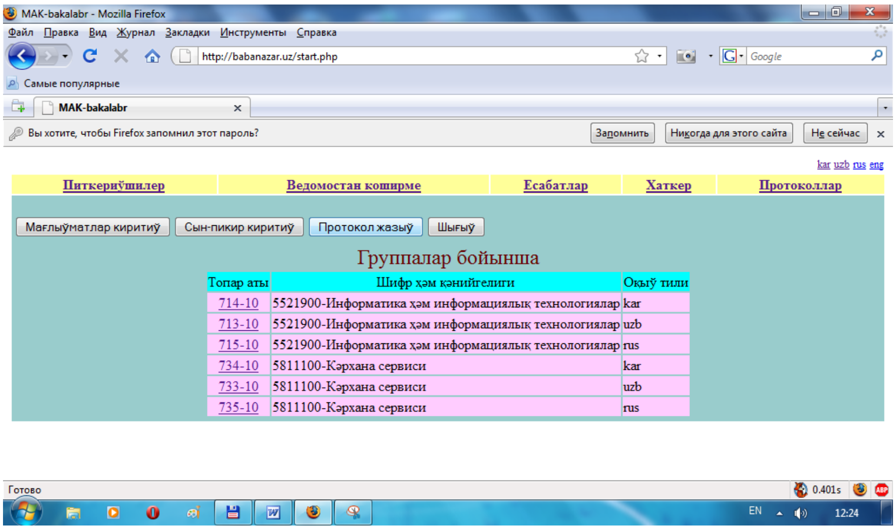
Бул веб-сайт көринисинде болған система Мәмлекетлик аттестация комиссиясының жумысларын автоматластырыўға арналған болып, мағлыўматлар базасына студентлер, топарлар, илимий басшылар ҳәм

мәслаҳатшылар, ПҚЖ темалары, студентлердиң пәнлерден баҳалары алдынан киритилген болып, ПҚЖ қорғаў дәўиринде МАК тәрепинен берилген сораўлар ҳәм шығарылған қарарлар хаткер тәрепинен киритиледи.

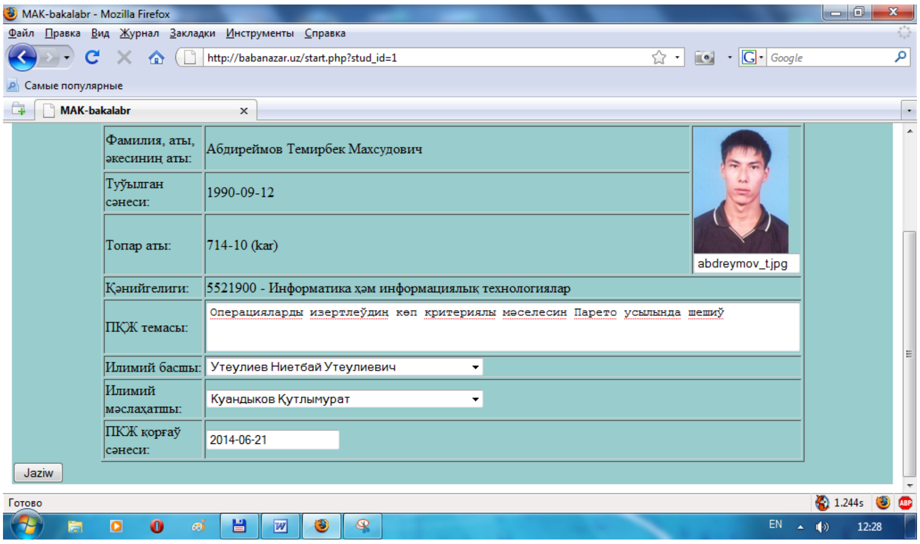
Хаткер өзиниң бөлимине кириў ушын пароль ҳәм логиннен пайдаланады.



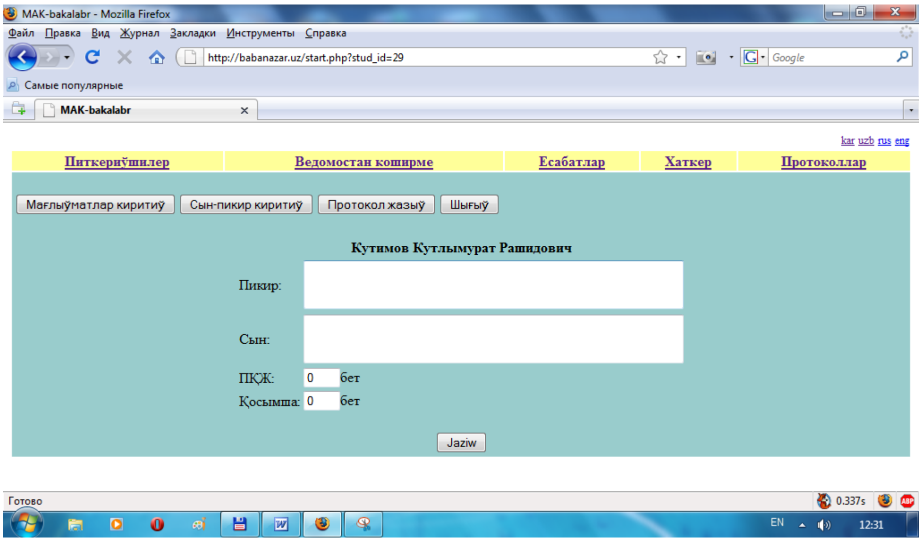
Логин ҳәм парол дурыс терилсе төмендеги айна пайда болады.



«Мағлыўматлар киритиў» арқалы биз студентти сайлап оған ПҚЖ темасын, илимий басшыны, илимий мәслаҳатшыны ҳәм деканат тәрепинен шығарылған қорғаў графиндеги студенттиң сәнесин киритиўимиз мүмкин.



«Сын-пикир киритиў» арқалы студенттиң ПҚЖ на берилген сын ҳәм пикирдиң мазмунуын ҳәмде коммисияға тапсырылған текст варианты көлеми ҳаққындағы мағлыўматларды киритип кетиўге болады.



«Протокол жазыў» жәрдеминде хаткер қорғаў ўақтында комиссия тәрепинен студентке берилген сораўларды ҳәм МАК тың қарарын киритиўге болады.

Булардан басқа система бизге, МАК ҳәр-бир студенттиң оқыған дәўири бойынша рейтингин көриўи, тийисли хызметкерлер ушын МАК тың барысы бойынша есабатлардың берилиўи, протоколлардың жазылыўының жеңиллесиўи, студентлердиң интернет арқалы МАК қарарын реал ўақытта алыўын береди.

Бул система тийкарынан студент әтирапына бирлестирилген, ҳәм бирлестирилетуғын сызықлы емес структурадағы мағлыўматлар тийкарында жумыс исеўге арналған болып, келешекте оны еледе кеңейтириўге болады. Оның ушын төмендеги жумыслар исплениўи керек:

1. Студентлерди баҳалаўдың электрон түрине өтиў керек.
2. Кафедрада ПҚЖ жумысларына илимий басшылар оқыў жылының басында бекитилгеннен кейин, системада илимий басшылар өзлери темалардың дәслепки вариантларын усыныўы керек.

3. Ҳәр-бир питкериўши студент системаға өзлериниң логин-пароли арқалы кирип өзлерине илимий басшы (басшыларды) сайлаў мүмкиншилигине ийе болыўы, ямаса илимий басшыға өзи қәлеген бағдардағы тема бойынша усыныс билдире алыўы, кейнинен еки тәреп келисими арқалы тема сайланып илимий кеңеске усынылыўы керек.
4. Илимий кеңес ағзалары ҳәр-бир темаға даўыс бериўи нәтийжесинде тема тастыйқланыўы ямаса қайта усыныў ушын көрип шығыўға илимий басшы ҳәм студентке қайтарылыўы керек.
5. Бул қатнасларды ислеў ушын ҳәр-бир системаның пайдаланыўшысына жеке кабинет ашылыўы керек.

Келешекте системаны бундай раўажланырыў МАК тың жумысын жеңиллестирип қоймай, ПҚЖ ның таярланыў жумысларына жеңиллик туўдырып, сыпатлылығын асырыўга хызмет етеди деп ойлайман.

ЖУЎМАҚЛАЎ

Компьютер технологияларының турмысымыздың дерлик барлық тараўларына кирип келиўи, олардың ислеўиниң тийкары болған программаластырыў технологияларынында тез пәт пенен раўажланыўы, программалардың қайта ислеўши дереги болған мағлыўматлардың қурамаласыўына алып келип, программаластырыў обектлери мағлыўматларын қолланыў, сақлаў, бөлистириў шешиў усылларының раўажланыўына үлкен тәсир етпекте.

Бул питкериў қәнигелик жумыста сызықлы емес структурадағы мағлыўматлар тийкарында программаластырыўда мағлыўматлар базаларынан пайдаланыў қаралып төмендеги мәселелер шешилди.

- Мағлыўматлар структуралары ҳәм олар үстинде орынланатуғын әмеллер үйренилип шығылды.
- Сызықлы емес структуралы мағлыўматлар ҳәм оларды программаластырыўда қолланыўда исленген мысаллар жәрдеминде танысылып өтилди.
- Программаластырыўда сызықлы емес структурадағы мағлыўматларды пайдаланыўда мағлыўматлар базалары таблицаларынан пайдаланыў усыллары үйренилди.
- Анық мысал ретинде ПҚЖ ларын қорғаўда МАК тәрепинен исленетуғын жумысларды автоматластырыў қаралып, мағлыўматлар базасы ҳәмде байланыслары исленип шығылды.
- Системаны жаратқанда тиллер мәселеси PHP тили ҳәм мағлыўматлар базасы MySQL жәрдеминде шешилди.

Бул исленген жумыслар нәтийжесинде сызықлы емес структурадағы мағлыўматларды программаластырыўда қолланғанда мағлыўматлар базалары жәрдеминен пайдаланыўда жақсы нәтийжелерге ерисиўге болатуғынлығы көрсетилди.

ПАЙДАЛАНҒАН ӘДЕБИЯТЛАР

1. А.А. Ключарев, В.А.Матьяш, С.В.Шекин Структуры и алгоритмы обработки данных – С.Петербург - 2004.
2. Непейвода Н.Н. Стили и методы программирования–М.Ижевск – 2004.
3. Вирт Н. Алгоритмы и структуры программы. М., Мир, 1985.
4. Берзтисс А.Т. Структуры данных. М. Статистика, 1984.
5. Трамбле Ж., Соренсон П. Введение в структуры данных. М., Машиностроение, 1982.
6. Ленгсам Й. Структуры данных для персональных ЭВМ. М., Мир, 1989.
7. Лойко В.И. Структуры и алгоритмы обработки данных. Учебное пособие для вузов.- Краснодар: КубГАУ. 2000. - 261 с., ил.
8. Орлов А.А.
PHP. Полезные приёмы. Издательство: Горячая -Телеком, 2004
9. Sh.A. Nazirov, R.V. Qobulov.
MySQL va PHP (qo'llanma). Uzbekistan 2008)
10. Томсон Л., Веллинг Л. Разработка Web-приложений на PHP и MySQL. - К.: "ДиаСофт", 2001.
11. Ҳамдамов Р., Парпиев А., Марахимов А., Бегимкулов У., Бекмурадов М., Н.Тайлоқов, Электрон университет. Масофавий таълим технологиялари. Олий таълим муассасалари учун. ЎзМЭ давлат илмий нашриёти. –Т.: 2008
12. http://www.php.ru/
13. http://www.intuit.ru/
14. http://www.allday2.com/
15. http://www.mysql.ru/docs/man/Connecting-disconnecting.html

ҚОСЫМША

Index.php файлы система биринши жүклениўин тәмийнлейди.

```
<?php
session_start();
include("configdb.php"); CONNECT_TO_MYSQL();
mysql_query('set names utf8');
?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<style type="text/css">
<!--
.style4 {font-size: 9px}
.style5 {font-size: 12px}
-->
</style>
</head>


<?
 $mk_m=0;


 $t="SELECT * FROM `menyu`";
 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);
 if ($mk_m<$r['id']){$mk_m=$r['id'];}
   $menyu_name[$r['id']]=$r['name'];
   $menyu_id_podsh[$r['id']]=$r['id_podsh'];
```

```
    $menyu_nomer_pp[$r['id']]=$r['nomer_pp'];
    $menyu_fail[$r['id']]=$r['faili'];}



  $t="SELECT * FROM `yazik`";
  $q=mysql_query($t);
  $n=mysql_num_rows($q);
  $kol_yazik=1;
  for ($i=0; $i<$n; $i++)
  {$r=mysql_fetch_array($q);
    $lang_nn[$r['nn']]=$r['name'];
  if ($kol_yazik<$r['nn']) {$kol_yazik=$r['nn'];}
 }
$tlang=$lang_nn[1];
$t_id=1;

  $_SESSION['tlang'] = $tlang;
  $_SESSION['kol_yazik'] = $kol_yazik;
  $_SESSION['lang_nn'] = $lang_nn;
  $_SESSION['menyu_id'] = $menyu_id;
  $_SESSION['menyu_name'] = $menyu_name;
  $_SESSION['menyu_id_podsh'] = $menyu_id_podsh;
  $_SESSION['menyu_fail'] = $menyu_fail;
  $_SESSION['menyu_nomer_pp'] = $menyu_nomer_pp;
  $_SESSION['mk_m'] = $mk_m;
  $_SESSION['t_id'] = $t_id;

?>
 <meta http-equiv="Refresh" content="0; url=start.php">
```

start.php файлы системаның модуллерин қурастырады.

```
<?
```

```
session_start();
$tstr='start.php';
?>


<script src="js/jquery-1.3.2.min.js" type="text/javascript"></script>
<script type="text/javascript">
$(document).ready(function(){
    $('.topmenu ul li').hover(
        function() {

            $(this).find('ul:first').stop(true, true);
            $(this).find('ul:first').slideDown();
        },
        function() {
            $(this).find('ul:first').slideUp('fast');
        }
    );
    $('.topmenu li:has(ul)').find('a:first').append('');
});


</script>

<style type="text/css">
.as1 {
	font-size: 24px;
	color: #660000;
}
a:focus {
	outline: none;
}
.topmenu {
    float:left;
}
.topmenu ul, .topmenu li{
    list-style:none;
```

```
  margin: 0;
  padding: 0;
}
.topmenu ul li {
	position:relative;
	width:220px;
}
.topmenu ul li a{
	display:block;
	padding:5px 5px 5px 5px;

	color: #0000FF;
	text-decoration: none;
	border:1px solid #999;
	margin:1px;
  background-color:#66FFFF;
  font-size: 14px;

}


.topmenu ul li a:HOVER{
	color: #00FF00;
  background-color:#FF66FF;
}

.topmenu ul li ul {
  display: none;
  position:absolute;
  top:0px;
  left:220px;
}

.ut-blue-button {
 display: -moz-inline-box;
 -moz-box-orient: vertical;
 display: inline-block;
```

```
  vertical-align: middle;
  *vertical-align: auto;
  background: url("blue-but.png") 0 0 no-repeat;
  color: white;
  text-decoration: none;
  height: 1px;
  padding-left: 1px;
  font: 12px/29px "Trebuchet MS", sans-serif;
  text-shadow: 0 1px 0 #1d7aaa; }
  .ut-blue-button {
    *display: inline; }
  .ut-blue-button span {
    cursor: pointer;
    cursor: pointer;
    display: -moz-inline-box;
    -moz-box-orient: vertical;
    display: inline-block;
    vertical-align: middle;
    *vertical-align: auto;
    background: url("blue-but.png") right -29px no-repeat;
    padding-right: 17px; }
    .ut-blue-button span {
      *display: inline; }
  .ut-blue-button:hover {
    background-position: 0 -58px; }
    .ut-blue-button:hover span {
      background-position: right -7px; }

</style>
</head>




<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
```

```
<head>
<title>MAK-bakalabr</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<link type="text/css" href="style.css" rel="stylesheet" media="all" />
<style type="text/css">
<!--
.style1 {color: #FFCCFF}
.style2 {font-size: 36px;font-weight: bold;color: #330066;font-style: italic;}
.style3 {font-size: 26px;font-weight: bold;color: #330066;font-style: italic;}
.style4 {font-size: 9px}
.pst {font-size: 0px}
.style5 {font-size: 12px}
.style6 {font-size: 22px;font-weight: bold;color: #330066;font-style: italic;}
.style7 {font-size: 26px;font-weight: bold;color: #0000FF;font-style: italic;}
-->
</style>
</head>
<head>
<body background="snow.gif">

<?
include ('configdb.php');
CONNECT_TO_MYSQL();
mysql_query('set names utf8');


$t="SELECT * FROM `stroki`";
  $q=mysql_query($t);
  $n=mysql_num_rows($q);
  for ($i=0; $i<$n; $i++)
    {$r=mysql_fetch_array($q);
     $perem[$r['perem']][$r['lang']]=$r['stroka'];}

#    $perem[$r['perem']][rus][lat]=$perem[$r['perem']][rus][kir];


if ($tlang<>$iz_lang and $iz_lang<>''){
```

```
    $_SESSION['tlang'] = $iz_lang;}


if ($t_id<>$iz_id and $iz_id<>''){

    $_SESSION['t_id'] = $iz_id;}


 echo '<table align="right" >';


    echo '<tr align=center style="font-size:12px;color:#0a4691" >';


    for ($j=1; $j<$kol_yazik+1; $j++){
        echo '<td colspan=2 align=center>';
        echo "<a href=".$tstr."?iz_lang=".$lang_nn[$j].">".$lang_nn[$j]."</a>";
        echo '</td>';
    }
echo '</tr></table>';



echo '<br>';

?>

<table width=100% ><tr bgcolor="#FFFF99">




<?
$kkbagana=0;

    for ($i=1; $i<$mk_m+1; $i++){
```

```
    if ($menyu_id_podsh[$i]==0 ){
        $kkbagana=$kkbagana+1;
    echo '<td align=center>';
    echo '<a
href='.$tstr.'?iz_id='.$i.'><strong>'.$perem[$menyu_name[$i]][$tlang].'</strong><
/a>';
    echo '</td>';}}

echo '</tr> <tr bgcolor="#99CCCC"><td colspan='.$kkbagana.'>';

if ($menyu_fail[$t_id]<>''){
include($menyu_fail[$t_id]);} ?>
</td>

</tr>
</table>


</body>




<?php
$skf='start.php';

$t="SELECT * FROM `gruppa`";
  $q=mysql_query($t);
  $km=mysql_num_rows($q);
  $max_gr_id=0;
  for ($i=0; $i<$km; $i++)
  { $r=mysql_fetch_array($q);
  if ($r['id']>$max_gr_id){$max_gr_id=$r['id'];}
 $gr_ati[$r['id']]=$r['name'];
 $gr_qan_id[$r['id']]=$r['qaniy_id'];
 $gr_bas_data[$r['id']]=$r['bas_data'];
 $gr_oqiw_tili[$r['id']]=$r['oqiw_tili'];}
```

```
$t="SELECT * FROM `qaniygelik`";
 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);
$qk_shifr[$r['id']][$r['lang']]=$r['shifr'];
$qk_name[$r['id']][$r['lang']]=$r['name'];
$qk_name_kor[$r['id']][$r['lang']]=$r['name_kor']; }

 $t="SELECT * FROM `student`";
   $max_st_id=0;
 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);
 if ($max_st_id<$r['id']){$max_st_id=$r['id'];}
$stu_surname[$r['id']]=$r['surname'];
$stu_name[$r['id']]=$r['name'];
$stu_lastname[$r['id']]=$r['lastname'];
 $stu_gruppa_id[$r['id']]=$r['gruppa_id'];
 $stu_foto[$r['id']]=$r['foto'];
 $stu_ts_data[$r['id']]=$r['ts_data'];
 }

 $t="SELECT * FROM `pkj_jumis`";

 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);
 $pkj_tema[$r['id_st']]=$r['tema'];
 $pkj_id_ilbas[$r['id_st']]=$r['id_ilbas'];
     $pkj_id_ilmas[$r['id_st']]=$r['id_ilmas'];
       $pkj_zash_data[$r['id_st']]=$r['zash_data'];}

$t="SELECT * FROM `ilm_bm`";
```

```
$q=mysql_query($t);
$km=mysql_num_rows($q);
for ($i=0; $i<$km; $i++)
{ $r=mysql_fetch_array($q);
$ilm_surname[$r['id']]=$r['surname'];
$ilm_name[$r['id']]=$r['name'];
$ilm_lastname[$r['id']]=$r['lastname'];
$ilm_ts_data[$r['id']]=$r['ts_data'];
$ilm_jinisi[$r['id']]=$r['jinisi'];
$ilm_mek_id[$r['id']]=$r['mek_id'];
$ilm_kb_id[$r['id']]=$r['kb_id'];
$ilm_lawazim[$r['id']]=$r['lawazim'];
$ilm_ildar[$r['id']]=$r['il_dar_id'];
$ilm_il_ataq[$r['id']]=$r['il_ataq']; }


$t="SELECT * FROM `kaf_bolim`";

$q=mysql_query($t);
$km=mysql_num_rows($q);
for ($i=0; $i<$km; $i++)
{ $r=mysql_fetch_array($q);
$kbl_name[$r['id']][$r['lang']]=$r['name'];
 }

 $t="SELECT * FROM `mekeme`";

$q=mysql_query($t);
$km=mysql_num_rows($q);
for ($i=0; $i<$km; $i++)
{ $r=mysql_fetch_array($q);
$mekeme_name[$r['id']][$r['lang']]=$r['name'];
 }

 $t="SELECT * FROM `il_dareje`";
```

```
  $q=mysql_query($t);
  $km=mysql_num_rows($q);
  for ($i=0; $i<$km; $i++)
  { $r=mysql_fetch_array($q);
  $ildar_name[$r['id']][$r['lang']]=$r['name'];
  $ildar_sokrname[$r['id']][$r['lang']]=$r['name_sokr'];
   }



 echo '<br>';
 echo '<br>';
echo '<table align=center width="80%" border=1 >';
$st_fio[1]=$stu_surname[$stud_id];$st_fio[2]=$stu_name[$stud_id];$st_fio[3]=$st
u_lastname[$stud_id];
        $fio_st=implode(" ", $st_fio);
echo '<tr>';
echo '<td height=50 width=120>'.$perem[fio][$tlang].':</td>';
echo '<td>'.$fio_st.'</td>';
echo '<td width="50" rowspan="3"><img src="suwretler/'.$stu_foto[$stud_id].'"
width="106" height="141"></td>';
echo '</tr>';

echo '<tr>';
echo '<td height=45>'.$perem[tuw_sane][$tlang].':</td>';
echo '<td>'.$stu_ts_data[$stud_id].'</td>';
echo '</tr>';

echo '<tr>';
echo '<td>'.$perem[grati][$tlang].':</td>';
echo '<td>'.$gr_ati[$stu_gruppa_id[$stud_id]].'
('.$gr_oqiw_tili[$stu_gruppa_id[$stud_id]].')</td>';
echo '</tr>';

echo '<tr>';
echo '<td>'.$perem[qaniygeligi][$tlang].':</td>';
```

```
echo '<td
colspan="2">'.$qk_shifr[$gr_qan_id[$stu_gruppa_id[$stud_id]]][$tlang].' -
'.$qk_name[$gr_qan_id[$stu_gruppa_id[$stud_id]]][$tlang].'</td>';
echo '</tr>';

echo '<tr>';
echo '<td>'.$perem[pqj_temasi][$tlang].':</td>';
echo '<td colspan="2">'.$pkj_tema[$stud_id].'</td>';
echo '</tr>';

echo '<tr><td>'.$perem[ilmiy_basshi][$tlang].':</td>';
echo '<td colspan="2">
'.$ildar_sokrname[$ilm_ildar[$pkj_id_ilbas[$stud_id]]][$tlang].'';
if ($ilm_il_ataq[$pkj_id_ilbas[$stud_id]]<>''){echo ', ';}
echo $ilm_il_ataq[$pkj_id_ilbas[$stud_id]];
echo ' '.$ilm_surname[$pkj_id_ilbas[$stud_id]].'
'.$ilm_name[$pkj_id_ilbas[$stud_id]].'
'.$ilm_lastname[$pkj_id_ilbas[$stud_id]].' -
'.$mekeme_name[$ilm_mek_id[$pkj_id_ilbas[$stud_id]]][$tlang].'
'.$kbl_name[$ilm_kb_id[$pkj_id_ilbas[$stud_id]]][$tlang].'
'.$ilm_lawazim[$pkj_id_ilbas[$stud_id]].'
</td></tr>';

echo '<tr>';
echo '<td>'.$perem[ilmiy_masl][$tlang].':</td>';
echo '<td colspan="2">
'.$ildar_sokrname[$ilm_ildar[$pkj_id_ilmas[$stud_id]]][$tlang].'';
if ($ilm_il_ataq[$pkj_id_ilmas[$stud_id]]<>''){echo ', ';}
echo $ilm_il_ataq[$pkj_id_ilmas[$stud_id]];
echo ' '.$ilm_surname[$pkj_id_ilmas[$stud_id]].'
'.$ilm_name[$pkj_id_ilmas[$stud_id]].'
'.$ilm_lastname[$pkj_id_ilmas[$stud_id]].' -
'.$mekeme_name[$ilm_mek_id[$pkj_id_ilmas[$stud_id]]][$tlang].'
'.$kbl_name[$ilm_kb_id[$pkj_id_ilmas[$stud_id]]][$tlang].'
'.$ilm_lawazim[$pkj_id_ilmas[$stud_id]].'
</td></tr>';
```

```
echo '<tr>';
echo '<td>'.$perem[qor_san][$tlang].':</td>';
echo '<td colspan="2">'.$pkj_zash_data[$stud_id].'</td>';
echo '</tr>';

echo '<tr>';
echo '<td>'.$perem[mak_qarari][$tlang].':</td>';
echo '<td colspan="2"></td>';
echo '</tr>';

echo '</table>';

?>



<?
include("configdb.php"); CONNECT_TO_MYSQL();
mysql_query('set names utf8');
session_start();
?>

<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">


<?

$tlang='kar';
$t="SELECT * FROM `qaniygelik`";
  $q=mysql_query($t);
  $km=mysql_num_rows($q);
  $max_kan_id=0;
  for ($i=0; $i<$km; $i++)
```

```
 { $r=mysql_fetch_array($q);
 if ($max_kan_id<$r['id']){$max_kan_id=$r['id'];}
$qk_shifr[$r['id']][$r['lang']]=$r['shifr'];
$qk_name[$r['id']][$r['lang']]=$r['name']; }

$t="SELECT * FROM `bloklar`";
 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 $max_blok_id=0;
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);
 if ($max_blok_id<$r['id']){$max_blok_id=$r['id'];}
$blok_name[$r['id']][$r['lang']]=$r['name']; }


 $t="SELECT * FROM `panler`";
 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 $max_pan_id=0;
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);
 if ($max_pan_id<$r['id']){$max_pan_id=$r['id'];}
$pan_name[$r['id']][$r['lang']]=$r['name']; }

$t="SELECT * FROM `oqiw_reje`";
 $q=mysql_query($t);
 $km=mysql_num_rows($q);
 for ($i=0; $i<$km; $i++)
 { $r=mysql_fetch_array($q);

$summ[$r['id_blok']][$r['id_qan']]=$summ[$r['id_blok']][$r['id_qan']]+$r['max_sa
at']; }


if ($dsor==''){
echo "<form  method=post action=ors.php>";
echo 'РљР°РЅРёР№РіРµР»РёРєС‚Рё С‚Р°Р»Р°РЅ';
```

```
echo '<select name=kaniy_id>';
for ($i=0; $i<$max_kan_id+1; $i++)
{echo "<option value=".$i.">";
echo $qk_shifr[$i][$tlang];echo '-';
echo $qk_name[$i][$tlang];
echo "</option>"; }
echo '</select>';
echo "<input type=hidden name=dsor value=11>";
echo '<input  type=submit value=РљРёСЃРёСѓ>';
echo "</form>";}


if ($dsor==11){
$_SESSION['dsor'] = $dsor;
$_SESSION['kaniy_id'] = $kaniy_id;

echo $qk_shifr[$kaniy_id][$tlang];echo '-';
echo $qk_name[$kaniy_id][$tlang];
echo '<table border=1>';
echo '<tr><td>в„–</td><td>РџУ™РЅ Р°С‚С‹</td><td>РЎР°Р°С‚
РєУ©Р»РµРјРё</td></tr>';
for ($i=1; $i<$max_blok_id+1; $i++){

echo '<tr><td><strong>'.$i.'</strong></td>
<td><strong>'.$blok_name[$i][$tlang].'</strong>
</td> <td align=center><strong>'.$summ[$i][$kaniy_id].'</strong></td></tr>';
$kk=1;
 $t="SELECT * FROM `oqiw_reje` ORDER BY `id` ASC";
  $q=mysql_query($t);
  $km=mysql_num_rows($q);
  for ($k=0; $k<$km; $k++)
  { $r=mysql_fetch_array($q);
  if ($r['id_blok']==$i and $r['id_qan']==$kaniy_id){
  echo '<tr>';

echo '<td>'.$i.'.'.$kk.'</td>';
echo '<td>';
```

```
}

echo $pan_name[$r['id_pan']][$tlang];
echo '</td>';

echo '<td align=center>'.$r['max_saat'].'</td>';
echo '</tr>';
$kk=$kk+1;
  }}




echo '<tr>';

echo '<td>'.$i.'.'.$kk.'</td>';

echo "<form  method=get action=zapis_ors.php>";
echo '<td>';
echo '<select name=pann_id>';
for ($j=0; $j<$max_pan_id+1; $j++)
{echo "<option value=".$j.">";
echo $pan_name[$j][$tlang];
echo "</option>"; }
echo '</select>';
echo '</td>';

echo '<td align=center><INPUT TYPE=TEXT SIZE=7 NAME=max_saat></td>';
echo '<input type=hidden name=pan_blok value='.$i.'>';
echo '<td><input  type=submit value=РЎР°РєР»Р°Сѓ></td>';
echo "</form>";

echo '</tr>';


}
```

```
echo '</table>';}

?>
</head>
</html>
```