**Բովանդակություն**

**Ներածություն……………………………………………………………………………...........5**

**Խնդրի դրվածքը………………………………………………………………………………....6**

**1. Գլուխ 1: Օգտագործվող գործիքամիջոցների և միջավայրի նկարագրությունը**

**1.1 ASP.Net Core միջավայրի նկարագրությունը..................................................................7**

**1.2 .Net Framework – ի նկարագրությունը...............................................................................8**

**1.3 C# լեզվի նկարագրությունը..............................................................................................10**

**1.4 Microsoft SQL Server.............................................................................................................13**

1.4.1 Ֆունկցիոնալություն..........................................................................................................15

1.4.2 Ծրագրերի մշակում...........................................................................................................17

**1.5 AES:ալգորիթմի նկարագրությունը.................................................................................19**

1.5.1 Գաղտնագրում....................................................................................................................20

1.5.2 Բանալիների ստացում......................................................................................................21

1.5.3 Գաղտնավերծանում..........................................................................................................22

**1.6 RCA ալգորիթմի նկարագրությունը..................................................................................23**

**2. Գլուխ 2 Համակարգի աշխատանքի և ալգորիթմի նկարագրությունը**

2.1 Նույնականացում..................................................................................................................24

2.2 Գրանցում համակարգում....................................................................................................26

2.3 Լոկալ գաղտնագրում և վերծանում..................................................................................28

**Եզրակացություն........................................................................................................................30**

**3 . Գլուխ 3: Ճյուղային տնտեսագիտություն........................................................................31**

**4. Գլուխ 4: Բնապահպանություն..........................................................................................40**

**5. Գլուխ 5: Կենսագործնեության անվտանգություն.....................................................................44**

**Գրականություն ցանկ...............................................................................................................48**

**Ներածություն**

Գաղտնագրության համակարգի իմաստն այն է, որ գաղտնի տեղեկատվությունը քողարկվի այնպես, որ այն անմատչելի լինի անիրավասու անձանց: Անկախ այն բանից, պահպանված է տեղեկատվությունը համակարգչի մեջ, թե հաղորդվում է համացանցի միջոցով, այն ապահովում է այն, որ ոչ իրավասու անձինք չեն կարող հասկանալ տեղեկատվության բովանդակությունը: Գաղտնահամակարգերը կարողեն լինել համաչափ կամ անհամաչափ ։ Իրենց հերթին տարբերակվում է նաև հոսքային կամ հատվածային ալգորիթմեր:

**Հոսքային** գաղտնագրի դեպքում տեքստը գաղտնագրվում է առանձին տառերով կամ բիտերով։ Յուրաքանչյուր տարրի (տառի կամ բիտի) գաղտնագրման կանոնը որոշվում է հաղորդագրության մեջ նրա դիրքով։

Հոսքային գաղտնագրերը ի տարբերություն հատվածայինի պարզ ու արագ են։ Բացի դրանից եթե ստացված հաղորդագրության ինչ֊որ հատվածում 1 բիթ սխալ է ստացվել, դա չի ազդի հաղորդագրութան մյուս հատվածների ճիշտ վերծանման վրա։

Սա հարմար է օրինակ բջջային կապի գաղտնագրման ժամանակ։

**Հատվածային** գաղտնագրի դեպքում երկուական հաջորդականությունը բաժանվում է տրված չափով հատվածների, և յուրաքանչյուր հատված գաղտնագրվում է առանձին։

Գաղտնակայուն հատվածային գաղտնագրերը օժտված են այս հատկություններով՝

• **Դիֆուզիա**՝ տեքստի նույնիսկ 1 բիթի փոփոխությունը պետք է առաջացնի

անկանխատեսելի փոփոխություններ գաղտնագրված տեքստում։

• **Անկարգայնություն**՝ հատարկման ժամանակ հակառակորդը չպետք է հասկանա, որ գտնվում է ճիշտ բանալու կողքին։

• **Ամբողջականություն**` գաղտնագրված տեքստի յուրաքանչյուր նիշ պետք է

կախված լինի բանալու յուրաքանչյուր նիշից։

Հատվածային գաղտնագրերը ի տարբերություն հոսքայինի ավելի գաղտնակայուն են։

Հաշվի առնելով գաղտնահամակարգերի առավելությունները և թերությունները մշակվեց հետևայլ համակարգը, որտեղ առկա են և հոսքային և հատվածային ալգորիթմեր:

**Խնդրի դրվածքը**

Խնդիր է դրվել՝

* Ուսումնասիրել տեսաշարում շարժվող օբյեկտի հայտնաբերման և ուղղության հաշվարկման համար անհրաժեշտ ալգորիթմները, խնդրի լուծման մոտեցումները, առաջարկել ալգորիթմ, որի կիրառմամբ առաջադրված խնդրին կտրվի լուծում։
* Ստեղծել ծրագիր, որը հնարաորություն կնձեռի օգտատերերին հաղերդակցվել անվտանգ գախտնագրված միջավայրում․

**Գլուխ 1**

**Օգտագործվող գործիքամիջոցների և միջավայրի նկարագրությունը**

**1.1 ASP.Net Core միջավայրի նկարագրությունը**

ASP.NET-ը վեբ-ծրագրերի և վեբ-սերվիսների ստեղծման տեխնոլոգիա է։ Այն իր անվանումը ստացել է Microsoft ASP-ից որոնք թեկուզ բավականին տարբերվում էին մեկը մյուսից։ [Common Language Runtime](https://ru.wikipedia.org/wiki/Common_Language_Runtime) (CLR)-ի հիման վրա Microsoft-ն ամբողջությամբ վերակազմավորեց ASP.NET-ը։ Ծրագրավորաղները կարող են գրել կոդ ASP.NET-համար, գրեթե բոլոր ծրագրավորման լեզուներով, սկսած  [.NET Framework](https://ru.wikipedia.org/wiki/.NET_Framework) ([C#](https://ru.wikipedia.org/wiki/C_Sharp), [Visual Basic.NET](https://ru.wikipedia.org/wiki/VB.NET) և [JScript .NET](https://ru.wikipedia.org/wiki/JScript_.NET) : ASP.NET-ն ըստ արագության ունի առավելություն համեմատած սկրիպտային տեխնոլոգիայի հետ, քանի որ հենց սկզբում դիմելիս կոդը ենթարկվում է կոմպիլյացիայի, և տեղադրվում է հատուկ քեշում։ Հետագայում այն կկատարվի, առանց ժամանակ ծախսելու վերլուծման, օպտիմալացման վրա և այլն։

ASP.NET-ի առավելությունները նախկին ASP-ի

* + Կոդի կատարումն ընթանում է արագ, սխալների մեծ մասը մշակման ժամանակ արդեն իսկ ցուցադրվում է,
  + Բավականին բարելավվել է արդեն պատրաստ ծրագրի իրականացման ժամանակ սխալների ուղղումը, օգտագործելով try…catch բլոկները,
  + Օգտագործաղի ղեկավարման էլեմենտները (controls), թույլ են տալիս առանձնացնել հաճախ օգտագործված շաբլոնները, ինչպես օրինակ կայքի մենյուն,
  + Ղեկավարման էլեմենտների ընդլայնված հավաքածուն և դասերի(կլասսների) գրադարանները թույլ են տալիս արագ մշակել ծրագիրը,
  + ASP.NET-ը հենվում է ․NET-ի բազմալեզու հնարավորության վրա, որը թույլ է տալիս գրել կոդ  VB.NET, Delphi.NET, Visual C#, J# էջերում,
  + Հնարավորությունը կատարել ամբողջ էջի կամ նրա մի մասի քեշավորում, բարձրացնում է արտադրողականությունը,
  + Հնարավորություն կատարելու քեշավորում օգտագործվող էջերում,
  + Հնարավորություն առաձնացնելու վիզուալ մասն ու բիզնես-տրամաբանականը առանձին ֆայլերով («code behind»),
  + Հարցերի մշակման ընդլայնված մոդել,
  + Սերվերային էլեմենտների ղեկավարման ընդլայնված մոդել,
  + Ըստ իր արագության ASP.NET-ն ունի առավելություններ համեմատած այլ տեխնոլոգիաների հետ։

**1.2 .Net Framework – ի նկարագրությունը**

.NET հենքը ստեղծված է աջակցելու բարդ և խիստ բաշխված կոմպոնենտային ծրագրերի նախագծման և աշխատեցման պրոցեսներին: Նա հնարավորություն է տալիս համատեղ օգտագործել տարբեր ծրագրային լեզուներ, ինչպես նաև ապահովում է ծրագրերի անվտանգությունը, տեղափոխելիությունը և Windows ՕՀ-ի ծրագրավորման ընդհանուր մոդելը: Ծրագրավորողի տեսանկյունից .NET-ը իրենից ներկայացնում է կատարող միջավայր և բազային տիպերի ահռելի գրադարան: Կատարման միջավայրի մակարդակը կոչվում է լեզուների համընդհանուր կատարման միջավայր (Common Language Runtime-CLR): Այս համակարգը կառավարում է ծրագրերի կատարումը: CLR-ը հանդիսանալով .NET-ի բաղադրիչ մաս, ապահովում է բազմալեզու ծրագրավորում, ինչպես նաև ծրագրերի տեղափոխելիությունը և անվտանգ կատարումը: CLR-ի հիմնական խնդիրը դա .NET-ի տիպերի ավտոմատ հայտնաբերման, բեռնման և կառավարման գործընթացն է(ծրագրավորողի փոխարեն): CLR-ը նաև զբաղվում է մի շարք ցածր մակարդակի խնդիրներով, ինչպիսիք են` հիշողության կառավարումը, ծրագրի սպասարկումը, հոսքերի վերամշակմամբ և անվտանգության հետ կապված զանազան ստուգներով: .NET-ի մյուս բաղադրիչը դա ընդհանուր տիպերի համակարգն է(Common Type System) կամ կրճատ` CTS համակարգ: CTS-ի հատկորոշման մեջ ներկայացված է բոլոր հնարավոր տիպերի և ծրագրային կառուցվածքների ամբողջ նկարագարությունը, որոնց աջակցում է կատարող միջավայրը, այն, թե ինչպես են այդ էությունները միմյանց հետ համագործակցում, և այն, թե ինչպես կարող են նրանք հանդես գալ որպես .NET-ի մետատվյալ:

Կարևոր է հասկանալ , որ CTS- ում ցանկացած որոշված ֆունկցիա կարող է չաջակցվել .NET-ի որոշակի լեզուների հետ: Դրա համար գոյություն ունի համընդհանուր լեզուների հատկորոշում (Common Language Specifiaction) կամ կրճատ` CLS հատկորոշում, որում նկարագրված է միայն այն ընդհանուր տիպերի ենթաբազմությունը և ծրագրային կառուցվածքները, որոնց կարող են աջակցել .NET –ի ծրագրավորման բոլոր լեզուները:

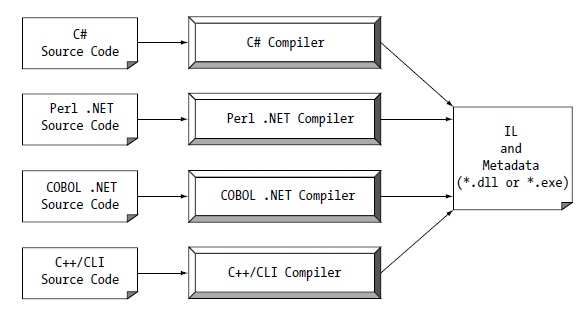
Ուստի .NET –ի միայն ֆունկցիոնալ հատկություններով տիպեր կառուցելու դեպքում, որոնք դիտարկված են CLS-ում, կարելի է լիովին վստահ լինել նրանում , որ .NET–ի հետ համատեղելի բոլոր լեզուները կարող են դրանք օգտագործել:Եվ հակառակը` CLS-ում չգտնվող որևէ տիպի տվյալների կամ ծրագրավորման կառուցվածքների օգտագործման դեպքում չի կարելի կարծել, որ .NET–ի հետ համատեղելի բոլոր լեզուները կարող են դրանք օգտագործել:



.NET-ի ծրագրավորման լեզուներից յուր­­­աքանչյուրն ընտրելիս, կարևոր է հիշել, որ .NET-ի երկուական կոդերի ֆայլերը ունեն նույն ընդլայնումները ինչ որ Win32-ի չկառավարվող ծրագրերի ֆայլերը(\* . dll կամ \* . ехе), բայց նրանց ներքին

կառուցվածքը տարբեր է: .NET-ի երկուական կոդերի ֆայլերը պարունակում են ոչ թե հատկորոշիչ տվյալներ այլ ընդհակառակը` հենքից անկախ տվյալներ, միջանկյալ

լեզվով(Intermediat Language -IL), ինչպես նաև տիպերի մետատվյալներ:



*Բոլոր .NET կոմպիլիատորները գեներացնում են IL կոդ և մետատվյալներ*

.NET-ի կոմպիլյատորի միջոցով \*.dll կամ \*.exe ֆայլ ստեղծելիս, ստացվում է մեծ երկուական օբյեկտ, որը կոչվում է հավաքածու (assembley): Այդ հավաքածուի մեջ է գտնվում CIL կոդը, որը հիմնովին նման է Java-ի բայթ կոդին նրանով, որ չի թարգմանվում հենքին հատուկ կոդի, մինչև չի լինում դրա անհրաժեշտությունը: Սովորաբար անհրաժեշտությունը լինում է, երբ CIL-կանոնների մի մասը դիմում է ֆունկցիայի, որը գտնվում է .NET-ի կատարող միջավայրում:

**1.3 C# լեզվի նկարագրությունը**

Քանի որ .NET-ը այդքան տարբերվում է իրեն նախորդող տեխնոլոգիաներից, Microsoft-ում ստեղծեցին հատուկ հենքի տակ աշխատող նոր ծրագրավորման լեզու` C#: C#-ը հանդիսանում է C ծրագրավորման լեզվի ընտանիքի անդամ դրա համար է, որ ունեն նման սինտակսիս:

Սինտակսիսի տեսանկյունից C#-ը հանդիսանում է մի քանի լեզվից ստեղծված գործնականում հզոր միջավայր, ինչպես C++-ը: Թվարկենք C# լեզվի մի քանի հատկություն՝

* Ցուցիչների կիրառման կարիք չկա: C# լեզվով գրված ծրագրերում սովորաբար ցուցիչներ կիրառելու պահանջ չի լինում (չնայած լեզուն հնարավորություն տալիս է օգտագործել ցուցիչներ):
* Հիշողության կառավարումը իրականացվում է ավտոմատ, այսպես կոչված` աղբ հավաքողի կողմից, դրա համար C#-ում չի կիրառվում delete հանգույցային բառը:
* Տրամադրվում են ֆորմալ սինտակսիսային կառուցվածքային տիպերի, ինտերֆեյսների, ստրուկտուրաների, թվարկությունների և դելեգատների համար:

.NET-ի տվյալ թողարկման մեջ C# լեզուն թարմացվել է, և օժտված է մի շարք նորություններով, որոնցից կարևորները թվարկված են ստորև`

* Ոչ անհրաժեշտ պարամետրերի հնարավորություն, ինչպես նաև անունով արգումենտներ:
* Անդամների դինամիկ որոնման հնարավորություն կատարման ընթացքում, dynamic բանալիային բառի միջոցով:
* Ընդհանրացված տիպերի հետ աշխատանքը բավականին հեշտացել է, ի շնորհիվ տվյալների հեշտ արտապատկերմա:

C#-ով ծրագրավորման ժամանակ կարելի է ստեղծել միայն այնպիսի կոդ, որը կթարգմանվի միայն .NET-ի միջավայրում:

**1.4 Microsoft SQL Server**

Microsoft SQL Server, ռելացիոն տվյալների բազաների կառավարման համակարգ ([DBMS](http://hy.wikipedia.org/w/index.php?title=DBMS&action=edit&redlink=1))` մշակված [Microsoft](http://hy.wikipedia.org/wiki/Microsoft) կորպորացիայի կողմից: Հարցումների օգտագործվող հիմնական լեզուն [Transact-SQL](http://hy.wikipedia.org/w/index.php?title=Transact-SQL&action=edit&redlink=1) է, որը ստեղծվել է համատեղ` Microsoft-ի և Sybase-ի կողմից։ Transact-SQL-ը հանդիսանում է ANSI/ISO ստանդարտի իրագործումը՝ ըստ հարցումների ենթակառուցվածքային լեզվի (SQL) ընդլայնումների։ Օգտագործվում է անհատականից մինչև հսկա կազմակերպությունների տվյալների բազաների հետ աշխատանքների համար, այն մրցակցում է շուկայի տվյալ սեգմենտում այլ ռելացիոն տվյալների բազաների կառավարման համակարգերի հետ։

**Պատմությունը**

MS SQL Server-ի ելակետային կոդը (մինչ 7.0 տարբերակը) հիմնվում էր [Sybase SQL Server](http://hy.wikipedia.org/w/index.php?title=Sybase_SQL_Server&action=edit&redlink=1) կոդի վրա, և դա թույլ տվեց Microsoft-ին՝ դուրս գալ կազմակերպությունների տվյալների բազայի շուկա, որտեղ այդ ժամանակ մրցակցում էին [Oracle](http://hy.wikipedia.org/w/index.php?title=Oracle&action=edit&redlink=1), [IBM](http://hy.wikipedia.org/wiki/IBM), իսկ ավելի ուշ ինքը՝ [Sybase](http://hy.wikipedia.org/w/index.php?title=Sybase&action=edit&redlink=1)-ը։ Microsoft-ը, [Sybase](http://hy.wikipedia.org/w/index.php?title=Sybase&action=edit&redlink=1)-ը և [Ashton-Tate](http://hy.wikipedia.org/w/index.php?title=Ashton-Tate&action=edit&redlink=1)-ը ի սկզբանե միավորվեցին՝ մի ծրագրի ստեղծման և շուկա բացթողնման համար, որը ստացավ [SQL Server 1.0](http://hy.wikipedia.org/w/index.php?title=SQL_Server_1.0&action=edit&redlink=1) OS/2 -ի համար անվանումը ([1989](http://hy.wikipedia.org/wiki/1989) թ.), որը փաստացի համարժեքն էր Sybase SQL Server 3.0 Unix, VMS-ի և այլնի համար։ Microsoft SQL Server 4.2-ը թողարկվել է 1992թվականին և գտնվում էր Microsoft OS/2 1.3 տարբերակի օպերացիոն համակարգի կազմում։ Microsoft SQL Server Windows NT օպերացիոն համակարգի համար 4.21 տարբերակի պաշտոնական հրապարակումը տեղի ունեցավ հենց [Windows NT](http://hy.wikipedia.org/w/index.php?title=Windows_NT&action=edit&redlink=1)-ի (տարբերակ 3.1) պաշտոնական հրապարակման (հանրայնացման) հետ միաժամանակ։

[Microsoft SQL Server 6.0](http://hy.wikipedia.org/w/index.php?title=Microsoft_SQL_Server_6.0&action=edit&redlink=1)-ը առաջին [SQL Server](http://hy.wikipedia.org/w/index.php?title=SQL_Server&action=edit&redlink=1)-ի տարբերակն էր, որը ստեղծված էր բացառապես Windows NT-ի ճարտարապետության համար և առանց մշակման գործընթացում Sybase-ի մասնակցության։

Այն ժամանակ, երբ շուկա դուրս եկավ Windows NT օպերացիոն համակարգը, [Sybase](http://hy.wikipedia.org/w/index.php?title=Sybase&action=edit&redlink=1)-ն ու Microsoft-ը բաժանվեցել էին և հետևում էին ծրագրային արտադրանքի սեփական մոդելներին և մարքեթինգային սխեմաներին։ Microsoft-ը ձգտում էր Windows-ի համար SQL Server-ների բոլոր տարբերակների գծով բացառիկ իրավունքի ձեռքբերմանը։ Ավելի ուշ, Sybase-ը փոխեց իր արտադրանքի անվանումը և անվանեց այն Adaptive Server Enterprise, խուսափելու համար Microsoft SQL Server-ի հետ շփոթելի լինելուց։ Մինչև 1994 թվականը Microsoft-ն արդեն ստացել էր երեք ծանուցում Sybase-ից՝ որպես ակնարկ Microsoft SQL Server-ի ծագման մասին։

Բաժանվելուց հետո ընկերությունները թողարկեցին մի քանի ինքնուրույն ծրագրեր։ SQL Server 7.0-ը դարձավ տվյալների բազայի առաջին սերվերը, որն օժտված էր օգտագործողի համար կառավարման իսկական գրաֆիկական ինտերֆեյսով։ Sybase-ի կողմից հեղինակային իրավունքի պահանջներից խուսափելու նպատակով 7-րդ տարբերակի ողջ ժառանգվող կոդը վերագրվեց։  
SQL Server 2005 տարբերակը ներկայացվել է 2005 թվականի նոյեմբերին։ Տարբերակի թողարկումն իրականացվում էր Visual Studio 2005-ի թողարկման հետ զուգահեռ։ Գոյություն ունի նաև Microsoft SQL Server-ի կրճատված տարբերակը՝ Microsoft SQL Server Express։ Այն հասանելի է բեռնման համար և կարող է անվճար տարածվել այն օգտագործող ծրագրային ապահովման հետ մեկտեղ։

SQL Server (SQL Server 2000)-ի նախորդ տարբերակի թողարկման պահից իրականացվում էին աշխատանքներ՝ մշակման ինտեգրացված ոլորտի և SQL Server 2005-ի կազմ մտնող մի շարք լրացուցիչ ենթահամակարգերի զարգացման ուղղությամբ։ Փոփոխություններից անմասն չմնացին նաև ETL (Տվյալների դուրսբերում, ձևափոխում և բեռնում) տեխնոլոգիայի իրացումը, որը մտնում էր SQL Server Integration Services (SSIS)-ի կազմի մեջ, ծանուցման սերվերը, տվյալների բազմաչափ մոդելների մշակման վերլուծական միջոցները (OLAP)-ը և անհրաժեշտ տեղեկատվության հավաքագրումը (երկու ծառայություններն էլ մտնում են Microsoft Analysis Services կազմի մեջ), ինչպես նաև մի քանի հաղորդագրության ծառայություններ, մասնավորապես՝ Service Broker և Notification Services։

Բացի այդ կատարվում էին աշխատանքներ նաև արտադրողականության բարելավման ուղղությամբ։

**1.4.1 Ֆունկցիոնալություն**

Տվյալների բազաների սերվերը Microsoft SQL Server որպես լեզվի հարցումներ օգտագործում է SQL լեզվի տարբերակը, որը ստացել է Transact-SQL (կրճատ T-SQL)։ T-SQL լեզուն [SQL-92](http://hy.wikipedia.org/w/index.php?title=SQL-92&action=edit&redlink=1)-ի (ISO ստանդարտ SQL լեզվի համար) իրականացումն է մի քանի ընդլայնումներով։ Պահվող պրոցեդուրաների համար T-SQL-ը թույլ է տալիս օգտագործել լրացուցիչ սինտաքսիս և ապահովում է գործարքի աջակցություն (կառավարման ծրագրով տվյալների բազաների փոխգործակցում)։

Microsoft SQL Server և Sybase ASE ցանցերի փոխգործակցության ժամանակ օգտագործվում է Tabular Data Stream (TDS, աղյուսակային տվյալների փոխանցման արձանագրություն) ծրագրային մակարդակի արձանագրություն։

TDS արձանագրությունը նույնպես իրականացվել է FreeTDS նախագծում, որպիսի տարբեր ծրագրերին տալ հնարավորություն փոխգործակցել Microsoft SQL Server և Sybase տվյալների բազաների հետ։

Տվյալներին մուտք ապահովելու համար Microsoft SQL Server-ը ապահովում է Open Database Connectivity (ODBC) – ծրագրերի փոխգործակվման ինտերֆեյս ՌՏԲԿՀ-ի հետ (Ռելացիոն տվյալների բազաների կառավարման համակարգ)։

SQL Server 2005 տարբերակը ապահովում է օգտագործողների վեբ-ծառայությունների միջոցով միացում՝ օգտագործելով SOAP արձանագրություն։ Այն թույլ է տալիս օգտագործողների ծրագրերին քրոսհարթակային կապով միանալ SQL Server-ին, նույնիսկ եթե ծրագրերը նախատեսված չեն Windows օպերացիոն համակարգի վրա։ Microsoft նույնպես թողարկել է հավաստագրված JDBC դրայվեր, որը թույլ է տալիս JAVA պլատֆորմի վրա աշխատող ծրագրերին (ինչպիսիք BEA и IBM WebSphere) կապվել Microsoft SQL Server-ին։

SQL Server-ը ապահովում է տվյալների բազաների հայլեացում և կլաստերզացիա։ SQL սերվերի կլաստերը նույն կոնֆիգուրացիա ունեցող սերվերների հավաքածու է. այդպիսի սխեման օգնում է աշխատանքի ծանրաբեռնվածությունը բաշխել մի քանի սերվերների միջև։ Բոլոր սերվերները ունեն միևնույն վիրտուալ անունը և տվյալները բաշխվում են կլաստերի IP հասցեների օգնությամբ, աշխատանքի փուլի ընթացքում։ Բացի այդ, մի սերվերի կլաստերի խափանման կամ անսարքության դեպքում հնարավոր է ծանրաբեռնման ավտոմատ փոխանցում ուրիշ սերվերին։

SQL սերվերը ապահովում է հավելյալ տվյալների կրկնապատկումը 3 սցենարներով.

Նկար. Ստեղծվում է տվյալների բազաների «նկարը», որը սերվերը ուղարկում է ստացողներին

Փոփոխումների պատմություն. Տվյալների բազաների բոլոր փոփոխությունները անընդհատ փոխանցվում են օգտագործողներին

Ուրիշ սերվերների հետ համաժամացում(սինխրոնիզացիա). Մի քանի սերվերների տվյալների բազաները համաժամացվում են միմյանց միջև։ Բոլոր տվյալների բազաների փոփոխությունները յուրաքանչյուր սերվերում կատարվում են միմյանցից անկախ, իսկ համաժամացման ժամանակ բոլոր տվյալները համեմատվում են։ Այդպիսի կրկնապատկման տեսակը ապահովում է տվյալների բազաների միջև տարաձայնությունների լուծման հնարավորությունը։

[SQL Server 2005](http://hy.wikipedia.org/w/index.php?title=SQL_Server_2005&action=edit&redlink=1)-ում ներդրված է [.NET Framework](http://hy.wikipedia.org/w/index.php?title=.NET_Framework&action=edit&redlink=1)-ի ապահովում։ Դրա շնորհիվ տվյալների բազաների պահպանելի պրոցեդուրաները կարող են գրվել .NET պլատֆորմի ցանկացած լեզվով՝ օգտագործելով գրադարանների ամբողջ հավաքածուն, որը առկա է .NET Framework-ի համար, գումարած Common Type System-ը (Microsoft .NET Framework-ում տվյալների տեսակների դիմելու համակարգ)։ Սակայն, ուրիշ պրոցեսների հետ համեմատած, .NET Framework-ը, լինելով բազիսային համակարգ SQL Server 2005-ի համար, հատկացնում է հավելյալ հիշողություն և օգտագործում է SQL Server-ի կառավարման միջոցները, Windows-ի միջոցները օգտագործելու փոխարեն։ Դա բարձրացնում արտադրողականությունը՝ համեմատած Windows-ի ընդհանուր ալգորիթմների հետ, քանի որ ռեսուրսների բաշխման ալգորիթմները հատուկ կարգավորված են SQL Server-ի կառուցվածքներում օգտագործվելու համար։

**1.4.2 Ծրագրերի մշակում**

Microsoft-ը և մյուս ընկերությունները ստեղծում են մեծ քանակությամբ մշակման միջոցներ, որը թույլ է տալիս, օգտվելով Microsoft SQL Server-ի տվյալների բազաներից, ստեղծել բիզնես-ծրագրեր։ Microsoft SQL Server 2005-ը նույնպես իր մեջ ներառում է Common Language Runtime (CLR) [Microsoft .NET](http://hy.wikipedia.org/w/index.php?title=Microsoft_.NET&action=edit&redlink=1), որը թույլ է տալիս իրականացնել պահպանվող պրոցեդուրաներ և տարբեր ծրագրային ֆունկցիաներ՝ ստեղծված .NET (օրինակ, [VB.NET](http://hy.wikipedia.org/w/index.php?title=VB.NET&action=edit&redlink=1) կամ [C#](http://hy.wikipedia.org/wiki/C)) լեզուների պլատֆորմի վրա։ Microsoft-ի նախորդ ստեղծված միջոցների տարբերակները օգտագործել են միայն API, որպիսի ստանան Microsoft SQL Server-ին ֆունկցիոնալ մուտք։

**SQL Server Express Edition**

[Microsoft SQL Server Express](http://hy.wikipedia.org/w/index.php?title=Microsoft_SQL_Server_Express&action=edit&redlink=1)-ը SQL Server տարբերակի անվճար տարածվող տեսակն է, [MSDE](http://hy.wikipedia.org/w/index.php?title=MSDE&action=edit&redlink=1) համակարգի զարգացմամբ։ Տվյալ տարբերակը ունի տեխնիկական որոշ սահմանափակումներ։ Այդպիսի սահմանափակումները դարձնում են նրան մեծ տվյալներ բազաների ստեղծման համար ոչ պիտանի, բայց այն միանգամայն պիտանի է փոքր ընկերությունների սահմաններում ծրագրային կոմպլեքսների ներմուծման համար։ Լիարժեք ապահովում է տվյալների նոր տիպերի օգտագործումը, այդ թվում [XML](http://hy.wikipedia.org/wiki/XML)-հատկորոշումներ (սպեցիֆիկացիա)։ Փաստորեն, այն լիարժեք MS SQL Server է, որը ներառում է իր ծրագրավորման բոլոր բաղադրիչները, ապահովում է ազգային և[Unicode](http://hy.wikipedia.org/wiki/Unicode) այբուբեններ։ Այդ պատճառով օգտագործվում է ծրագրերում, նախագծման կամ ինքնուրույն հետազոտման համար։ Չկա ոչ մի խոչընդոտ՝ հետագայում կուտակած MS SQL Server ոչ էքսպրես տարբերակի տվյալների բազաների տեղակայման համար։ [2007](http://hy.wikipedia.org/wiki/2007) թ. Microsoft-ը թողարկեց գրաֆիկական ինտերֆեյսով առանձին գործիք այդ տարբերակի կառավարման համար, որը նույնպես կարելի է անվճար բեռնել ընկերության պաշտոնական կայքից։

**Սահմանափակումներ**

1 պրոցեսոր(այն կարող է տեղադրված լինել ցանկացած սերվերում)

1 գբ հասցեային հիշողություն

4 գբ բազայի առավելագույն չափս (10 գբ SQL Server 2008 R2-ի համար)

SQL Server Management Studio 2005-ի ինտերֆեյսից անհնար է տվյալների մուտքագրում/արտածում (2008 տարբերակի մեջ այդ հնարավորությունը կա)

[Windows](http://hy.wikipedia.org/wiki/Windows)-ի համար նախատեսված տարբերակներ

1992 - SQL Server 4.2

1993 - SQL Server 4.21 Windows NT-ի համար

[SQL Server 2008](http://hy.wikipedia.org/w/index.php?title=SQL_Server_2008&action=edit&redlink=1)-ը ուղղված է տվյալների ղեկավարումը ինքնակարգավորվող, ինքնակառավարվող և ինքնասպասարկող մեխանիզմ դարձնելուն. այդպիսի հնարավորությունների իրականացման համար ստեղծվել են SQL Server Always On տեխնոլոգիաները: Այն հնարավորություն է ընձեռում զրոյացնել սերվերի չաշխատելու վիճակի ժամանակը:

SQL Server 2008-ում ավելացվել է կառուցվածքային և կիսակառուցվածքային տվյալների ապահովումը, ներառելով` նկարների, ձայնային, վիդեո և այլ թվային մուլտիմեդիա տիպի ֆորմատներ։ Մուլտիմեդիա-ֆորմատների ապահովումը բազաներում թույլ տվեց հատուկ ֆունկցիաների փոխգործակցել այդ տիպի տվյալների հետ։

Բացի այդ, ներառվեցին նաև հատուկ ֆորմատներ՝ ամսաթիվ և ժամանակ և տարածային (անգլ. Spatial) տիպ հատուկ տարածակախյալ տիպի տվյալների համար։ Ոչ կառուցվածքային տվյալների համար ստեղծվեց հատուկ տիպ. օրինակ՝ Պատկեր։

SQL Server-ում կառավարման էֆեկտիվությունը բարձրացնելու համար ներառվեցին [Declarative Management Framework](http://hy.wikipedia.org/w/index.php?title=Declarative_Management_Framework&action=edit&redlink=1) գրադարանները, որի շնորհիվ հնարավոր է բաշխել լիազորություններ տվյալների բազաների կամ առանձին աղյուսակների համար։ Նորացվել են տվյալների խտացման եղանակները։ [. SQL Server Katmai](http://hy.wikipedia.org/w/index.php?title=._SQL_Server_Katmai&action=edit&redlink=1)-ն[[1]](http://hy.wikipedia.org/wiki/Microsoft_SQL_Server#cite_note-2k8-1)[[2]](http://hy.wikipedia.org/wiki/Microsoft_SQL_Server#cite_note-PaulFlessner-2)ապահովում է [ADO.NET Entity Framework](http://hy.wikipedia.org/w/index.php?title=ADO.NET_Entity_Framework&action=edit&redlink=1) գրադարանների հավաքածուն և կրկնման ծանուցումների միջոցներ ու տվյալների տիպի որոշում։

**AES գաղտնագրում**

Գաղտնագրման համար AES ալգորիթմում կիրառվում են տվյալների ձևափոխության հետևյալ գործընթացները՝

1. ExpandKey - Բոլոր փուլերի համար փուլային բանալիների հաշվում:

2. SubBytes - Փոխարինման աղյուսակի միջոցով բայթերի փոխարինում:

3. ShiftRows - Վիճակում տողերի, տարբեր քանակությամբ, ցիկլիկ տեղաշարժ:

4. MixColumns - Վիճակի յուրաքանչյուր սյունյակում տվյալների խառնում:

5. AddRoundKey - Փուլի բանալու գումարումը վիճակին:

Գաղտնագրումը տեղի է ունենում նկ. 1-ում պատկերված ալգորիթմով:

**SubBytes ձևափոխությունը**

SubBytes ձևափոխությունը կայանում է վիճակի ամեն {xy} բայթը (որտեղ x և y 16-ական համակարգի թվեր են), աղ. 1-ի համաձայն, մեկ քայլով փոխարինման մեջ:

**ShiftRows** **ձևափոխությունը**

ShiftRows ձևափոխությունը իրենից ներկայացնում է վիճակի տողերի ցիկլիկտեղաշարժ դեպի ձախ: Առաջին տողը մնում է նափոփոխ, երկրորդում կատարվում է տեղաշարժ մեկ բայթով, երրորդում 2 բայթով, չորրորդում՝ 3:

**MixColumns ձևափոխությունը**

MixColumns ձևափոխության ժամանակ կատարվում է 4-րդ աստիճանի քառակուսային մատրիցի բազմապատկում վիճակի յուրաքանչյուր սյունյակի հետ: Յուրաքանչյուր սյունյակի հետ գործողությունը կատարվում է առանձին:

**AddRoundKey** **ձևափոխությունը**

AddRoundKey ձևափոխության ժամանակ փուլային բանալու 32-բիթանոց բառերը XOR բիթային գործողության միջոցով գումարվում են վիճակի սյուներին: Յուրաքանչյուր սյունյակի հետ գործողությունը կատարվում է առանձին:

**ExpandKey** **գործընթացը**

AES ալգորիթմում փուլային բանալիները գեներացվում են գաղտնագրման բանալու հիման վրա՝ ExpandKey գործընթացի միջոցով: ExpandKey գործընթացը ստեղծում է Nb \* (Nr + 1) բառ, որտեղ Nb-ն բլոկում բառերի քանակն է, Nr-ը՝ փուլերի քանակը: Ալգորիթմին անհրաժեշտ է Nb երկարությամբ սկզբնական բանալի, ինչպես նաև Nr փուլերից յուրաքանչյուրը պահանջում է Nb բառերից կազմված բանալի:

**AES վերծանում**

Վերծանման ժամանակ բոլոր ձևափոխությունները կատարվում են հակառակ հերթականությամբ: Գոյություն ունեցող գաղտնագրող ձևոփոխությունների փոխարեն օգտագործվում են հետևյալ հակադարձ ձևափոխությունները՝

1. InvSubBytes - Բայթերի փոխարինում հակադարձ փոխարինման աղյուսակի միջոցով,

2. InvShiftRows - Վիճակում տողերի ցիկլիկ տեղաշարժ տարբեր քայլերով,

3. InvMixColumns - Վիճակի յուրաքանչյուր սյունյակում տվյալների խառնում:

ExpandKey և AddRoundKey գործընթացները մնում են անփոփոխ: Փուլի բանալիները օգտագործվում են հակառակ հերթականությամբ: Վերծանման ալգորիթմը ներկայցաված է նկ. 2-ում:

**InvShiftRows ձևափոխությունը**

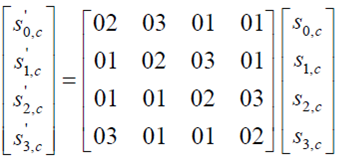
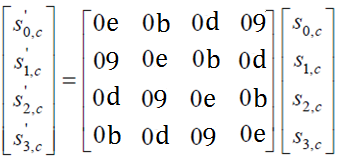
Սա ShiftRows ձևափոխությանը հակադարձ ձևափոխություն է: Վիճակի առաջին տողը մնում է անփոփոխ, երկրորդը 1 բայթով ցիկլիկ տեղաշարժվում է աջ, երրորդը 2 բայթով, չորրորդը՝ 3:

**InvSubBytes ձևափոխությունը**

Սա SubBytes ձևափոխությանը հակադարձ ձևափոխություն է: Բայթերի փոխարինումը իրականացվում է համանմանորեն՝ հակառակ փոխարինումների աղյուսակի համաձայն:

**InvMixColumns ձևափոխությունը**

Սա MixColumns ձևափոխությանը հակադարձ ձևափոխություն է: Վիճակի յուրաքանչյուր սյունյակը ձևափոխվում է առանձին:



InvMixColumns

ձևափոխությունը

MixColumns

ձևափոխությունը

**RSA գաղտնագրային ալգորիթմը**

RSA-ը հանդիսանում է գաղտնագրման առաջին լիակատար անհամաչափ ալգորիթմը, որը կարելի է օգտագործել ինչպես տեղեկատվության գաղտնագրված փոխանակման, այնպես էլ թվային ստորագրության ձևավորման համար և ստեղծվել է 1977թ.: Այն անվանվել է ալգորիթմը մշակոած երեք գիտնականների՝ Ռոն Ռիվեստի (Ron Rivest), Ադի Շամիրիր (Adi Shamir) և Լեոնարդ Էդլմանի (Leonard Adleman) անզգանունների սկզբնատառերով: Գոյություն ունեցող անհամաչափ գաղտնագրային ալգորիթմներից RSA-ն հավանաբար ամենահեշտն է հասկանալու և իրագործելու տեսակետից և միաժամանակ գործնականում ամենատարածվածը: Այս ալգորիթմը երկար տարիներ դիմակայում է ինտենսիվ գաղտնավերլուծությանը: Չնայած գաղտնավերլուծությունը ո՛չ ապացուցել է, ո՛չհակասել RSA-ի ավտանգությունը, այն, ըստ էության, հիմնավորում է ալգորիթմի հանդեպ վստահությունը:

RSA-ի անվտանգությունը հիմնված է մեծ թվերը բազմապատկիչների վերլուծելու բարդության մեջ: Բաց և փակ բանալիները հանդիսանում են երկու մեծ պարզ թվերի ֆունկցիաներ: Ենթադրվում է, որ ըստ գաղտնագրված տեքստի և բաց բանալու, բաց տեքստի վերականգնումը համարժեք է երկու մեծ թվերը բազմապատկիչների վերլուծելուն:

**RSA բանալիների ստացումը**

Երկու բանալիների գեներացման համար օգտագործվում են երկու մեծ պարզ թիվ՝ P և Q: Առավելագույն անվտանգության համար P-ն և Q-ն ընտրվում են միևնույն երկարության: Հաշվում ենք հետևյալ արտադրյալը.

N = P \* Q

Այնուհետև ընտրվում է 1-ից մեծ E թիվ, որը (P-1) \* (Q-1) արտադրյալից փոքր է և փոխադարձաբար պարզ է նրա հետ: N և E թվերը հանդիսանում են բաց բանալին: Վերջում Էվկլիդեսի ընդլայնված ալգորիթմով հաշվվում է D փակ բանալին.

D = E-1 mod ((P-1)(Q-1))

D և N թվերը նույնպես փոխադարձաբար պարզ են: P և Q թվերը այլևս պետք չեն, սակայն պետք է գաղտնի պահվեն, քանի որ նրանց միջոցով կարելի է հաշվարկել D փակ բանալին:

**RSA գաղտնագրում և վերծանում**

M հաղորդագրությունը գաղտնագրելու համար նախևառաջ այն բաժանվում է N-ից փոքր թվային բլոկների (երկուական տվյալների համար ընտրվում է N-ից փոքր 2-ի ամենամեծ աստիճանը): Այսինքն, եթե P-ն և Q-ն 100-կարգանի պարզ թվեր են, ապա N-ը կպարունակի մոտ 200 կարգ և հաղորդագրության յուրաքանչյուր Mi բլոկի երկարությունը պետք է լինի մոտ 200 կարգ: C գաղտնագրված հաղորդագրությունը բաղկացած կլինի միևնույն երկարությամբ Ci բլոկներից: Գաղտնագրման բանաձևն ունի հետևյալ տեսքը՝

ci = mie mod n

Հաղորդագրությունը վերծանելու համար պետք է վերցնել գաղտնագրված Ci բլոկներից յուրաքանչյուրը և հաշվել

mi = cid mod n

Քանի որ

cid = (mie(mod n))d = mied(mod n) = mik(p-1)(q-1)+1(mod n) =

= mi\*mik(p-1)(q-1) (mod n) = mi\*1(mod n) = mi (mod n)

հետևաբար բանաձևը վերականգնում է հաղորդագրությունը:

Նույն կերպ հաղորդագրությունը կարող է գաղտնագրվել D-ի միջոցով և վերծանվել E-ի միջոցով:

**RSA ալգորիթմի արագությունը**

RSA-ի գլխավոր թերությունը կայանում է շատ ցածր արագագործության մեջ: Այդ իսկ պատճառով, մեր խնդրում RSA ալգորիթմով գաղտնագրվում է միայն AES ալգորիթմի բանալին, իսկ տեքստը գաղտնագրվում է AES ալգորիթմով, այսինքն՝ ընդհանուր համակարգի արագագործությունը էականորեն չի տուժում:

RSA ալգորիթմով գաղտնագրում իրականացվում է զգալիորեն ավելի արագ, եթե E-ի արժեքը ճիշտ է ընտրվում: Ամենից հաճախ կիրառվող երեք տարբերակներն են՝ 3, 17 և 65537 (65537 թվեի երկուական ներկայացումները պարունակում է երկու հատ մեկ, ինչի շնորհիվ աստիճան բարձրացնելիս պետք է կատարել ընդամենը 17 բազմապատկում): Այս երեք թվերից ոչ մեկի կիրառության դեպքում չկա անվտանգության խնդիր (այն պայմանով, որ հաղորդագրությունը լրացվում է պատահական թվերով), նույնիսկ եթե E-ն օգտագործվում է օգատագործողների մի ամբողջ խմբի կողմից:

Փակ բանալիներով գործողությունները կարելի արագացնել նաև մնացորդների մասին չինական թեորեմի միջոցով, եթե պահպանվել են P, Q, արժեքները, ինչպես նաև d mod (p - 1), d mod (q - 1) և q-1 mod p լրացուցիչ արժեքները:

**RSA ալգորիթմի անվտանգությունը**

RSA-ի անվտանգությունը ամբողջությամբ կախված է մեծ թվերը բազմապատկիչների վերլուծելու խնդրից: Տեխնիկապես, անվտանգության մասին այս պնդումը կեղծ է: Երբևէ մաթեմատիկորեն չի ապացուցվել, որ ըստ C-ի և E-ի M-ը վերականգնելու համար պետք է N-ը վերլուծել բազմապատկիչների: Հասկանալի է, որ հնարավոր է հայտնաբերվի RSA-ի գաղտնավերլուծության ամբողջովին այլ մեթոդ: Սակայն, եթե այդ մեթոդը օգնի գաղտնավերլուծողին ստանալ D-ն, այն կարող է օգտագործվել նաև մեծ թվերը բազմապատկիչների վերլուծելու ժամանակ:

RSA-ն կարելի է կոտրվել նաև գուշակելով (P-1)(Q-1) արժեքը, ինչն ավելի հեշտ չէ քան N-ը բազմապատկիչների վերլուծելը:

N-ը բազմապատկիչների վերլուծելը կոտրելու ամենաակնհայտ տարբերակն է: Ցանկացած հակառակորդ կարող է ունենալ E բաց բանալին և N մոդուլը: Վերծանման D բանալին գտնելու համար հակառակորդը պետք է N-ը վերլուծի բազմապատկիչների: Ներկա դրությամբ ամենամեծ թիվը, որը վերլուծվել է բազմապատկիչների պարունակում է 129 թվանշան՝ տասական համակարգով: Հետևաբար N-ը պետք է մեծ լինի այդ թվից:

Իհարկե, գաղտնավերլուծողը կարող է փորձել D-ի բոլոր հնարավոր տարբերակները, մինչև գտնի ճիշտ թիվը: Այսպիսի մոտեցումը նվազ արդյունավետ է քան N-ը բազմապատկիչների վերլուծելը:

Ժամանակ առ ժամանակ տարբեր հայտարարաություններ են հայտնվում այն մասին, իբր RSA-ը կոտրելու հեշտ մեթոդը գտնվել է, սակայն դեռևս այդ հայտարարություններից ոչ մեկը չի հաստատվել:

Կա RSA ալգորիթմի անհանգստության համար վստահ լինելու ևս մեկ առիթ: P և Q պարզ թվերի հաշվարկման ալգորիթմների մեծամասնությունը հավանականային են: Իսկ ի՞նչ կլինի, եթե P-ն և Q-ն լինեն բաղադրյալ: Առաջին հերթին կարելի է նման դեպքի հավանականությունը հասցնել նվազագույնի: Եվ նույնիսկ, եթե այդպիսի բան տեղի ունենա, ամենայն հավանականությամբ նմանատիպ դեպքը միանգամից կհայտնաբերվի՝ գաղտնագրումն ու վերծանումը չեն աշխատի: Գոյություն ունեն թվեր, որոնք չեն հայտնաբերվում պարզ թվերի որոմնան հավանականային ալգորիթմների կողմից: Այդ թվերը շատ հազվադեպ են և կոչվում են Կարլմայքլի թվեր:

**Գլուխ 2**

**Ծրագրի աշխատանքի և ալգորիթմի նկարագրությունը**

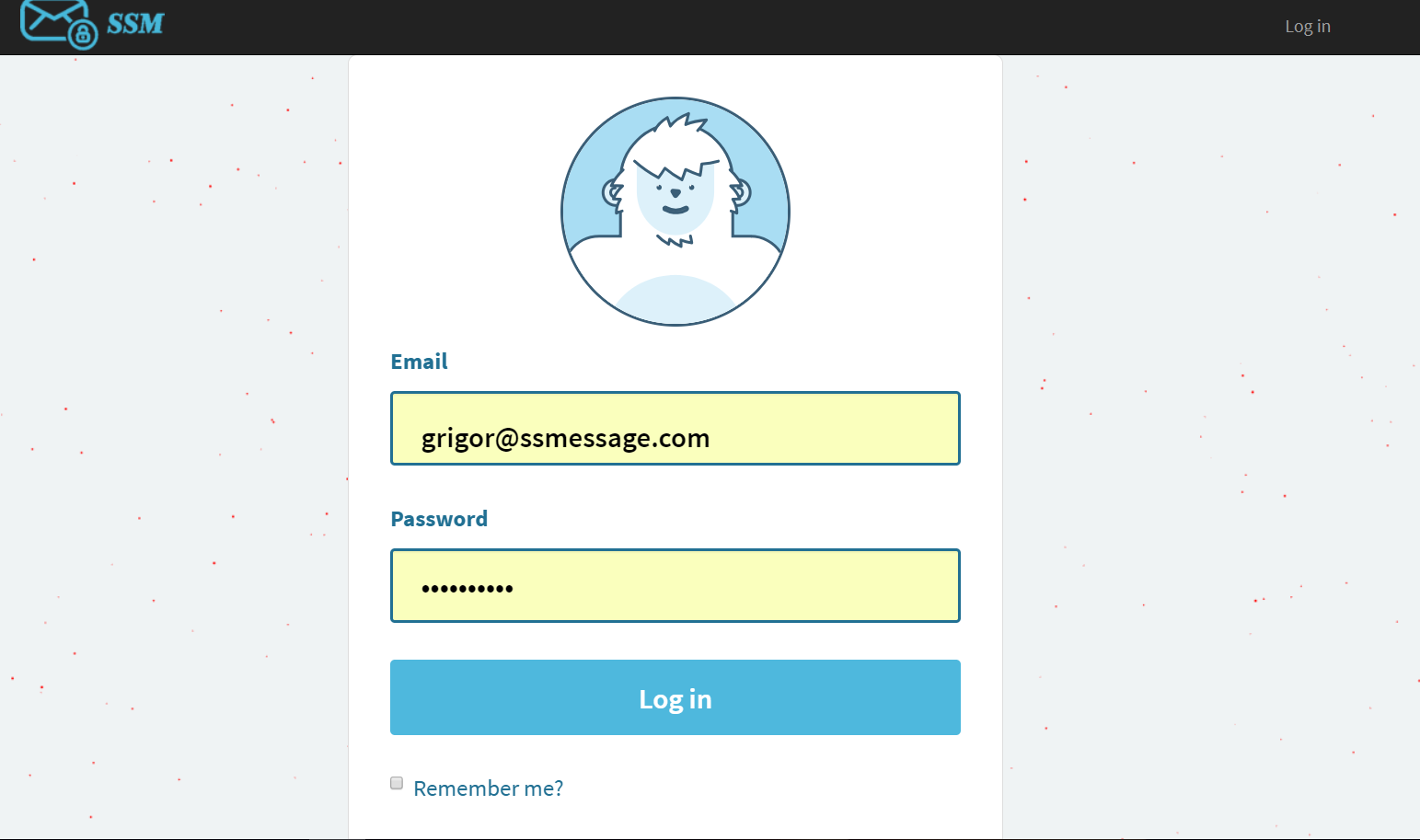
**2.1 Նույնականացում**

Հիմք ընդունելով նշանակված խնդիրները՝

* Ավտոմատացնել գաղտնագրման և վերծանման գործողությունները և բանալիների փոխանակումը
* Բաց կապուղիով երկու օգտատերի միջև անվտանգ փոխանցել տվյալները

Առաջարկվում է հետևայլ լուծումը. Ստեղծել ընդհանուր մեկ համակարգ որը կոգտագործի բաց կապուղի, միևնույն ժամանակ լինի անվտանգ և հեշտ կիրառելի ցանկացած ոլորտի օգտագործողի համար: Պետք է ավտոմատացված լինի բանալինների փոխանակումը երկու օգտագործողների միջև:

Դիտարկենք համակարգի սկզբնական պատուհանը:

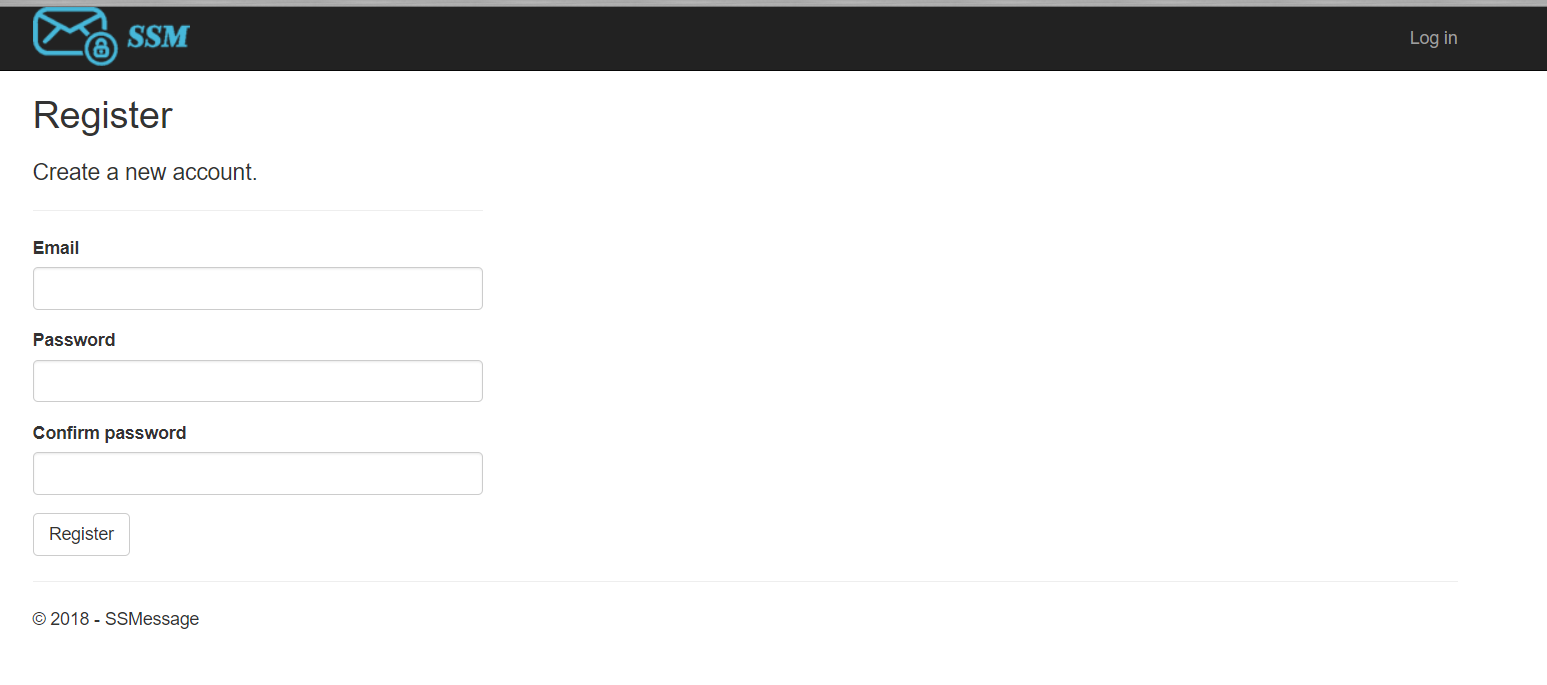


Նկ. 18 Համկարգի սկզբնական պատուհան

Համակարգ մուտք գործելու համար անհրաժեշտ է ունենալ նույնականացման տվյալներ: Որը մուտքագրելուց հետո համեմատում է տվյալների բազայում առկա տվյալների հետ և որոշում թույլ տա, թե` ոչ:

**2.2 Գրանցում համակարգում**

Համակարգում գրացվելու համար պետք է սեղմել Register կոճակը, որից հետո կբացվի գրացման պատուհանը (նկ. 19)



Նկ. 19 Գրանցման պատուհան

Գրանցման պատուհանի մեջ առկա են նույնականացման անհրաժեշտ մինիմալ պահանջները: Քանի որ բոլոր օգտատերերը լինելու են հավասար պայմաններում, երեք դաշտը բավական է անվտանգ նույնականացում կիրառել: Պետք է լրացնել Login, password և e-mail դաշտերը:

**Login**- դաշտը յուրահատուկ է լինելու բոլորի համար, այսինքն նա չի կրկնվի, որի շնորհիվ կարողանում ենք կիրառել նույնականացման ժամանակ:

**Password**-իհարկե գաղտնաբառի յուրահատկությունը ավելորդ է: Անվտանգությունը ապահովելու համար այն հեշավորվում է և պահվում սերվերում:

**E-mail-** էլեկտրոնային հասցեն պետք է լինի յուրահտւկ և օգտագործվում է գաղտնաբառը վերականգնելու համար:

Համակարգը ավտոմատ ստուգում է բոլոր յուրահատուկ պայմանները բազային մակարդակի վրա:

cmd.CommandText = "SELECT \* FROM [diplom].[dbo].[user] WHERE Login='" + this.textBox\_Login.Text + "' OR E-mail ='"+this.textBox\_email+"'";

Տվյալները լրացնելուց հետո պետք է սեղմել «Registration» և համակարգը հայտնում է գրանցումը հետևայլ նախադասությամբ- "Data Entered Successfully"

Համակարգում կիրառվում է SHA-512 հեշավորման ֆունկցիան:

**Հեշ ֆունկցիաները** պարզապես անհակադարձելի գաղտնագրեր են։

Հաղորդագրության հեշը նրա «խտացված պատկերն» է։

Հեշ ֆունկցիայի արգումենտը կարող է լինել կամայական երկարություն, իսկ նրա

արդյունքը ունի ֆիքսված երկարություն։

Հնարավոր է այնպիսի դեպք, երբ 2 արժեք ունեն նույն հեշը։ Այդ վիճակը կոչվում է

բախում (collision)։

Եթե գործնականում հնարավոր է գտնել մի արժեք, որի հեշը տալիս է որոշակի

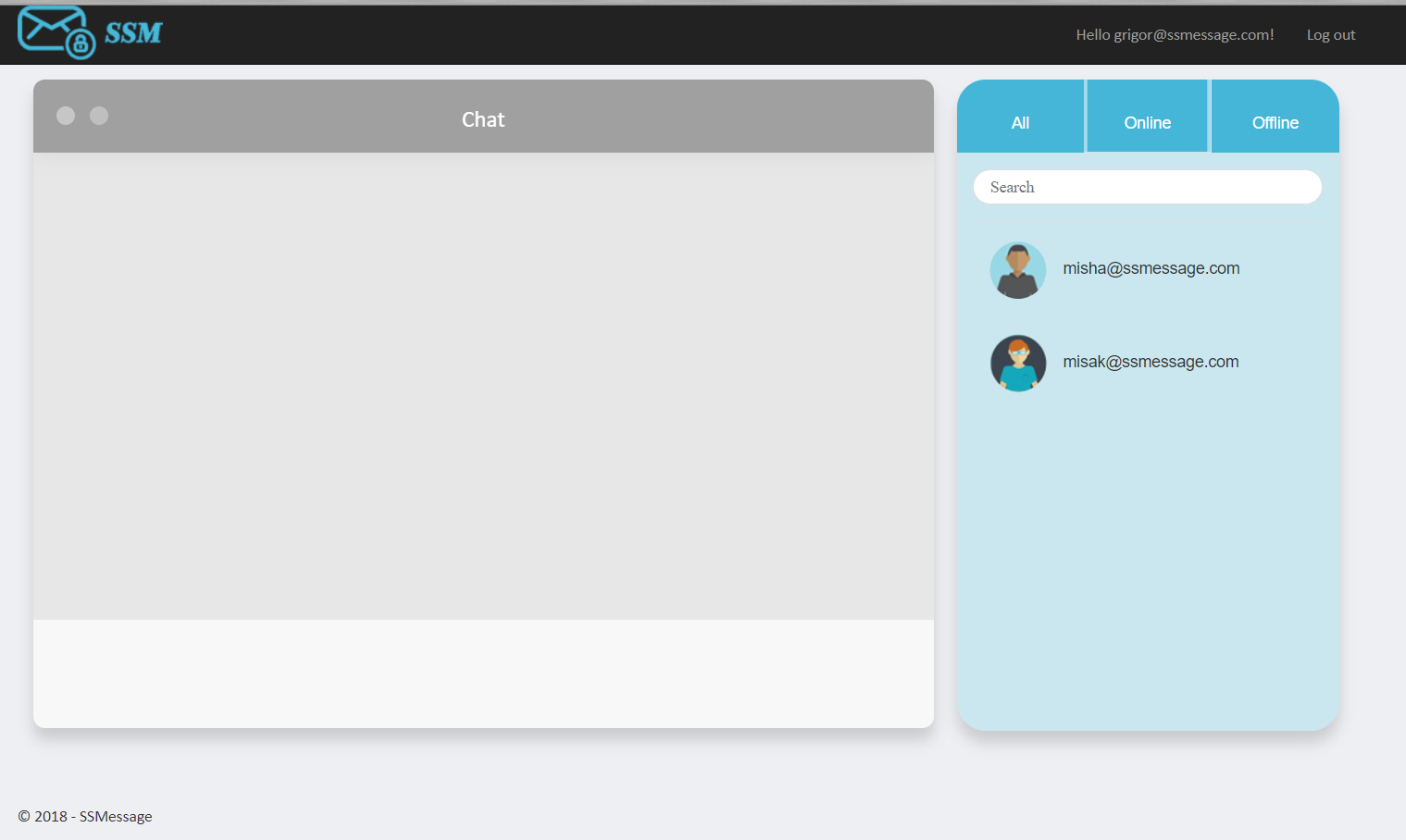
արժեք, ապա հեշը համարվում է կոտրված, և չպետք է օգտագործվի անվտանգության

ապահովման համար։

Հեշեր օգտագործվում են օր․՝ գաղտնաբառերի պահպանման ժամանակ, ֆայլերի

ամբողջականության ստուգման ժամանակ, և այլն։

Համակարգ մուտք գործելուց հետո բացվում է նկ. 20-ում պատկերված պատուհանը:

Նկ. 20 Հիմնական պատուհան

**2.3 Գաղտնագրում և վերծանում**

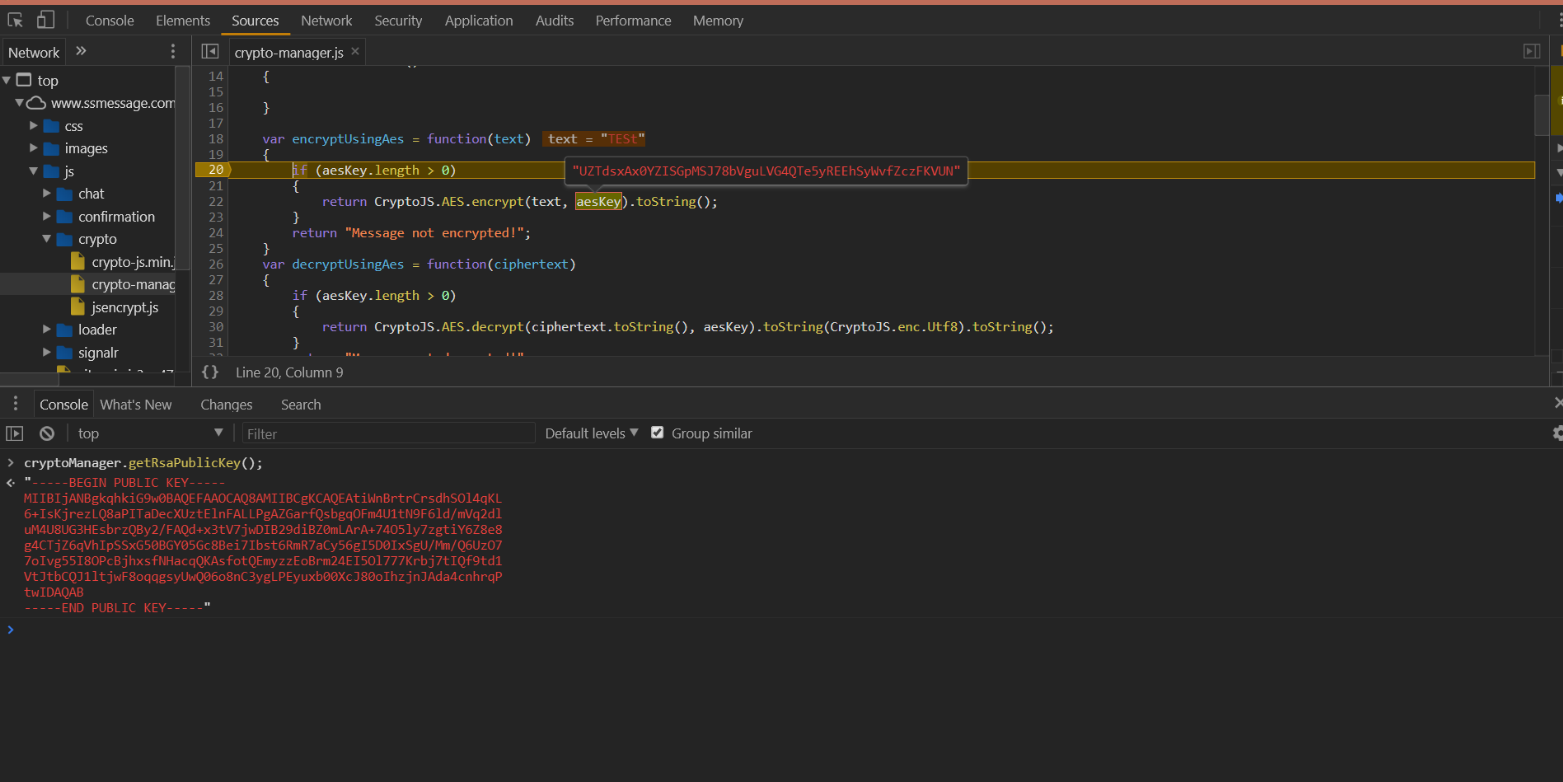
**Գաղտնագրումը** հնարավոր է կատարել JS (JavaScript)-ում դա մեզ հնարաորություն

Է տալիս ստեղծել բանալի, որը կգտնվի տվյալ օգտատիրոջ Browser-ի օպերատիվ հիշողության մեջ․SSM(Send Secure Message) –ում օգտագործվել է RSA ոչ համաչափ գաղտնագրման ալգորիթմը և AES՝ համաչափ գաղտնագրման ալգորիթմը․

RSA-ն օգտագործվում է ամեն նոր օգտատերի հետ հաղորդակցության սկսումից առաջ ստեղծվում է մեկ փակ մեկ բաց բանալի և բաց բանալին ուղղարկում է մյուս օգտատիրոջն, որն իր հերթին իր մոտ random գեներացնում է AES-ի համաչափ բանալի և ուղղարկում է հետ։

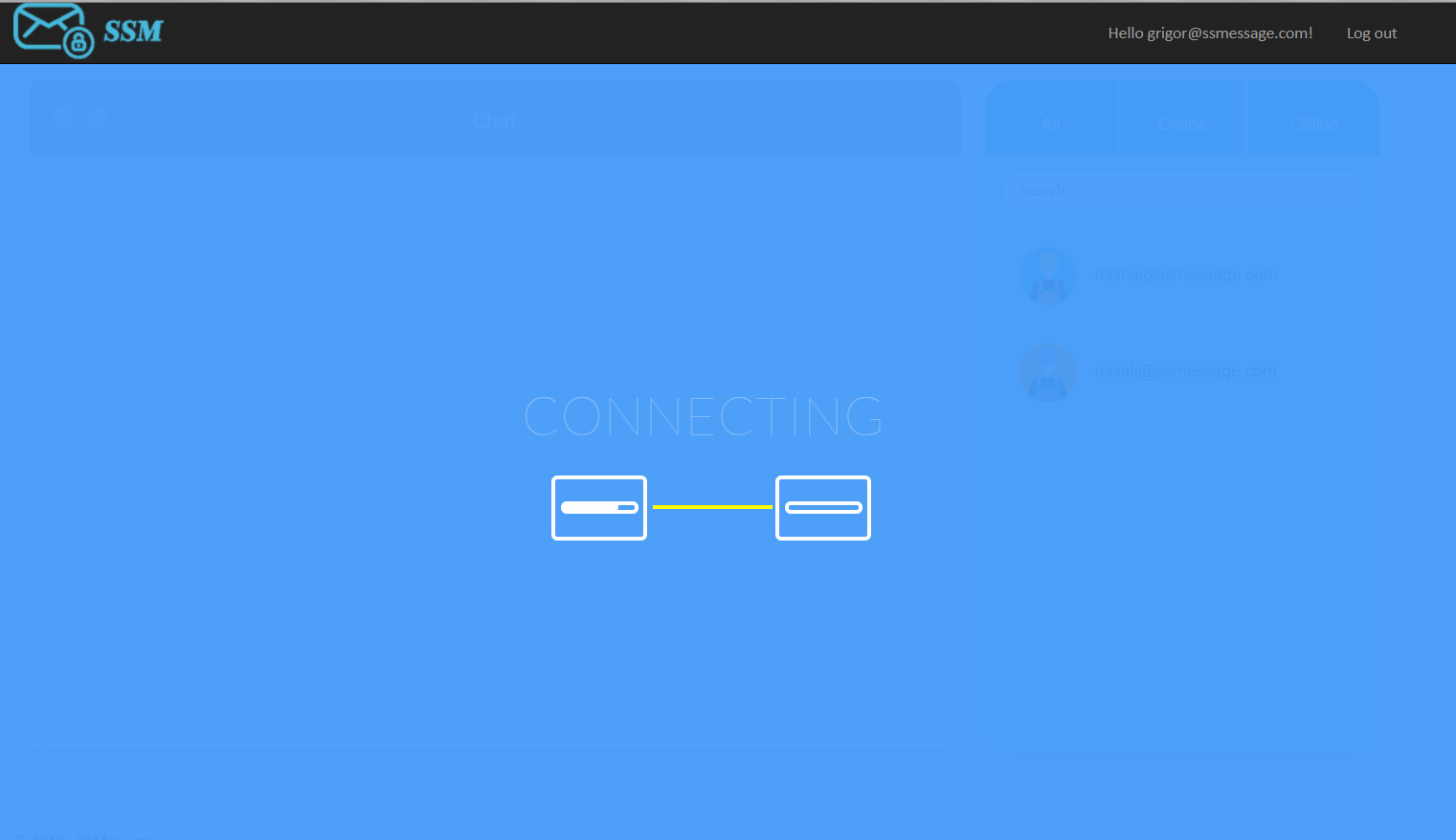
Օգտատերերը սկսում են իրենց հաղորդակցությունը՝ օգտագործելով AES-ի համաչափ գաղտնագրման ալգորիթմը։

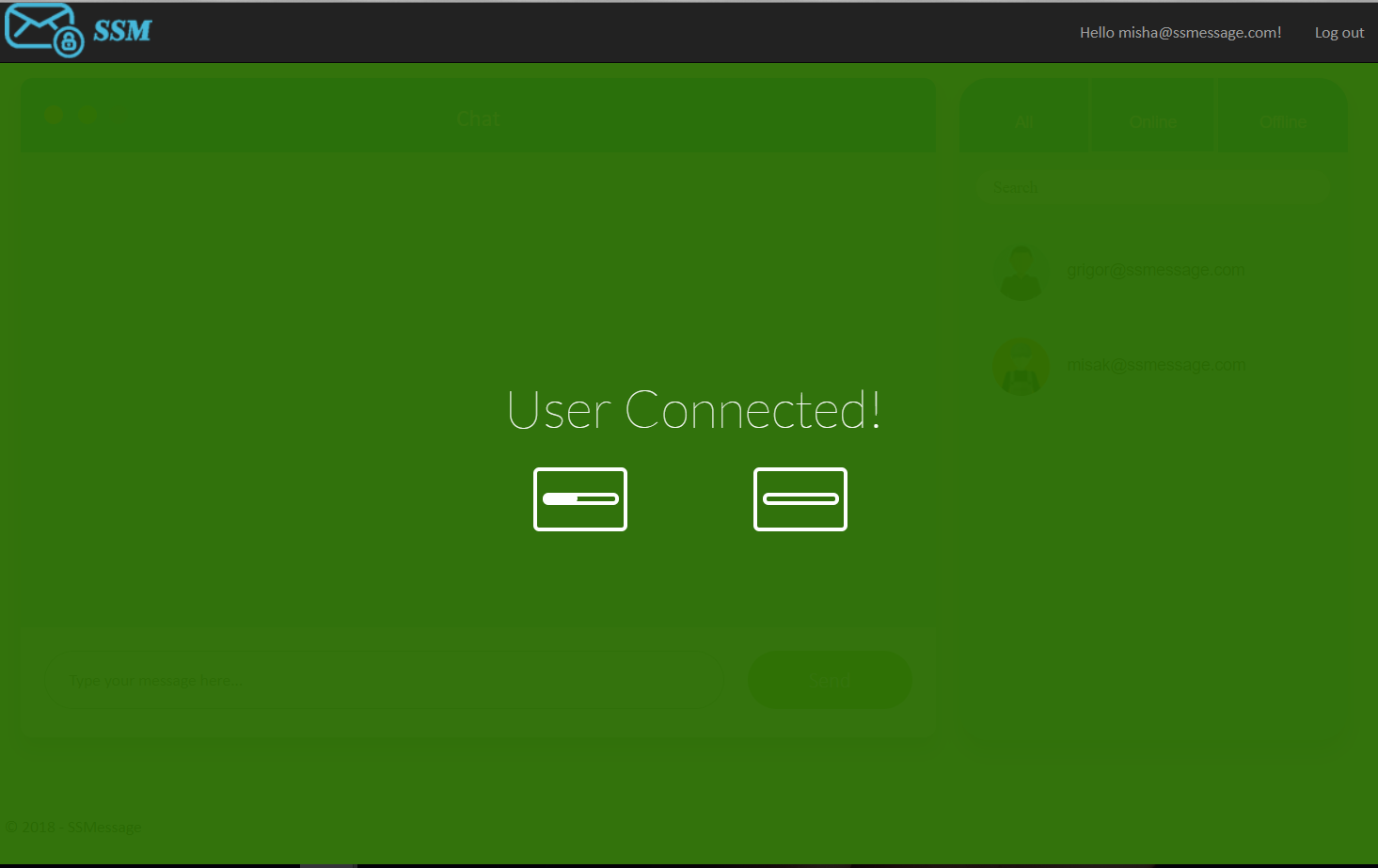
Նկ 21 ում կարող եք տեսնել AES –ի և RSA-ի բանալիները․



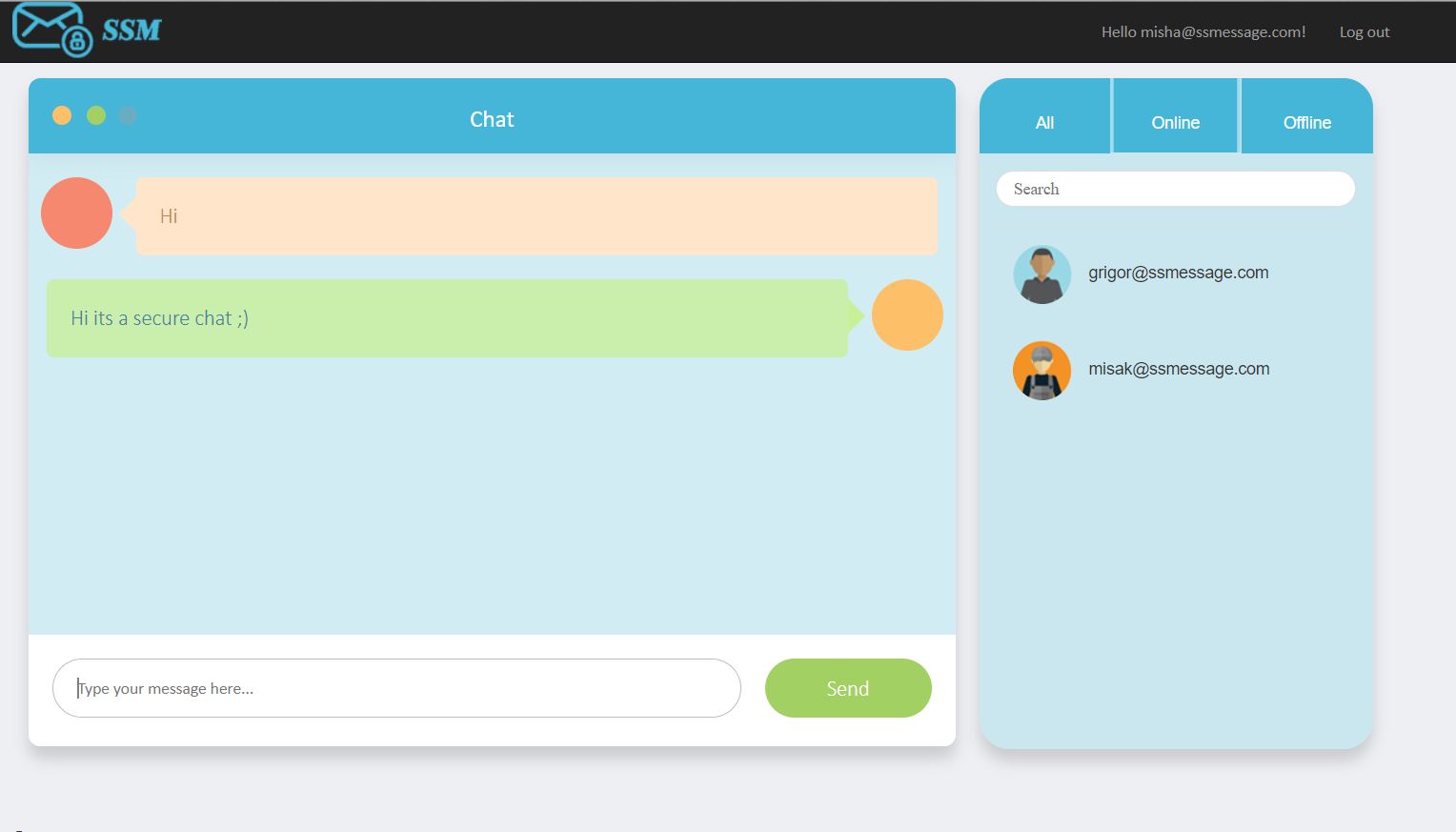
Նկ. 21 բանալիների պատուհան

**Վերծանման** վերծամումը կատարվում է համապատասխան բանալի ի օգնությամբ օգտագործելով AES ալգորիթմը տես՝ (նկ. 21):

 Նկ. 22 բանալիների փոխանակման գործնթաց



Նկ. 22 բանալիների փոխանակման բարեհաջող ավարտ



Նկ. 22 անվտանգ հաղորդակցություն

**Եզրակացություն**

Այսպիսով աշխատանքի ընթացքում ուսումնասիրվեցին WebSocket, SSLSecure Sockets Layer, նրանցում հաղորթագրության անվտանգ փոխանցման հնարավորությունները ինչպես նաև առկա մեթոդների դրական և բացասական կողմերը: Մշակվեց նոր համակարգ որն հնարաորություն է տալիս արանց SSL-ը Certeficate-ը ունենալու ունենալ բավականին անվտանգ հաղորթակցության միջոց, մենք ունեցանք գաղտնագրման և վերծանման գործողությունների ու բանալինների փոխանակման ավտոմատացում:

**Գլուխ 3**

Ավարտական աշխատանքի տնտեսական մասի

Հաշվարկներ

**SSM-(Send Secure Message) համակարգչային հավելվածի ծրագրային փաթեթի ինքնարժեքի և գնի հաշվարկը**

Ժամանակակից տնտեսական պայմաններում ցանկացած գործունեություն առնչվում է տնտեսական հարցերի հետ: Նոր տեխնիկայի, տեխնոլոգիական պրոցեսների, տեխնիկա-տնտեսական հիմնավորման համար, մասնավորապես մեր աշխատանքի պայմաններում կարևորագույն տնտեսական հիմնահարցերից է ինքնարժեքի որոշումը:

Արտադրանքի կամ ծառայությունների ինքնարժեքը` դա արտադրանքի (ծառայությունների) արտադրության և իրացման վրա կատարված բոլոր ծախսերի գումարն է դրամական արտահայտությամբ:

Ինքնարժեքի մեջ իրենց արտահայտությունն են գտնում սպառված շրջանառու ֆոնդերը, կենդանի աշխատանքի մի մասը, որը աշխատողներին վճարում է աշխատավարձի ձևով:

Ինքնարժեքի մեջ մտնող ծախսերը դասակարգվում են ըստ տնտեսական տարերի և ըստ կալկուլյացիոն հոդվածների:

Ներկայումս կիրառվում է ծախսերի ըստ կալկուլյացիոն հիմնական հոդվածների հետևյալ դասակարգումը`

1. համալրող առարկաներ,

2. էլեկտրաէներգիայի ծախսեր,

3. աշխատողների հիմնական աշխատավարձ,

4. աշխատողների լրացուցիչ աշխատավարձ,

5. սարքավորումների շահագործման և պահպանման ծախսեր,

6․ տարածքի վարձակալության համար ծախս,

7․ ընդհանուր տնտեսական ծախսեր։

**Լրիվ ինքնարժեք (1-7 կետերի գումարը)։**

Փաթեթը, որի ինքնարժեքն ու գինը ենթակա է որոշման, իրենից ներկայացնում է՝ համակարգչային հավելվածի ծրագրային փաթեթ:

Հաշվարկի համար ելքային տվյալներ են հանդիսանում`

- փաթեթի մեջ մտնող համալրող առարկաների քանակն ու անվանացանկը;

- ժամանակի ամփոփ նորմերը, աշխատանքի կարգն ու աշխատավարձի ձևերը,

- ժամավճարային և գործարքային պարգևատրման չափերը ( 24%),

- լրացուցիչ աշխատավարձի չափերը (18 %),

- սարքավորումների պահպանման և շահագործման ծախսերի դրույքաչափերը,

- ընդհանուր տնտեսական ծախսերի դրույքաչափը (117 %)։

1. **Համալրող առարկաների ծախսի հաշվարկ**

Գնված բաղադրիչների արժեքը որոշվում է հետևյալ բանաձևով`

Cпк = ∑Mпkj\*Pпkj,

որտեղ` Mпkj j-րդ տեսակի գնված բաղադրիչների քանակն է, հատ, Pпkj j-րդ գնված բաղադրիչի գինը, դրամ: Հաշվարկի արդյունքները բերված են աղյուսակ 1-ում:

*Աղյուսակ 1.*

|  |  |  |  |
| --- | --- | --- | --- |
| Բաղադրիչի անվանումն ու տեսակը | Մեկ փաթեթին ընկնող քանակ.հատ | Միավորի գինը.դրամ | Մեկ փաթեթին ընկնող արժեք.դրամ |
| Համակարգիչ | 5 | 300 000 | 1.500.000 |
| Սեղան աթոռ | 5 | 50 000 | 250.000 |
| Սպասք | 15 | 10.000 | 150.000 |
| Ինտերնետ | 2 | 20.000 | 40.000 |
| Ընդամենը | 27 | 38.000 | 1.940.000 |

Ընդամենը` 1.940.000 դրամ:

1. **Էլեկտրաէներգիայի ծախսի հաշվարկ**

Համակարգիչները և այլ սարքավորումները աշխատեցնելու համար էլեկտաէներգիայի տարեկան ծախսը որոշվում է հետևյալ բանաձևով`

**¾=W\*U¾**

որտեղ` W-ն էլեկտրաէներգիայի տարեկան ծախսն է, UԷ-ն 1 կվ/ժ էլեկտրաէներգիայի արժեքն է (≈50 դրամ)։

Էլեկտրաէներգիայի ամսեկան ծախսը մոտավոր կկազմի W≈200կՎտ/ժ, Է=200\*50=10000 դրամ/ամսական և 10000\*12=120000 դրամ/տարեկան:

1. **Աշխատողների հիմնական աշխատավարձի հաշվարկը**

Աշխատողների հիմնական աշխատավարձի մեջ մտնում են`

● գործարքային դրույքաչափերով աշխատավարձ,

● ժամավճարային աշխատավարձ,

● պարգևավճար:

Գործարքային աշխատավարձն ըստ տարիֆային համակարգի որոշվում է հետևյալ բանաձևով`

Ահիմ.= Ժդ.\*Աարտ.,

որտեղ` Ժդ.-ն ժամային դրույքաչափն է, Աարտ.-ն ՝ ժամային նորմը: Հաշվարկման արդյունքները բերված են աղյուսակ 2-ում:

*Աղյուսակ 2*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Գործառույթի հաջորդական. | Վճարման ձև | Ժամանա-կային նորմ | Ժամային դրույք | Տարիֆային ֆոնդ |
| Նախապատ-րաստում | Գործարքա-պարգևա-վճարային | 18 | 3500 | 63000 |
| Մշակում | Գործարքա-պարգևա-վճարային | 25 | 5000 | 125000 |
| Կարգավորում | Գործարքա-պարգևա-վճարային | 18 | 5000 | 90000 |
| Տեստավորում | Գործարքա-պարգևա-վճարային | 37 | 4500 | 166500 |
| Ընդամենը | 98 | | | 444500 |

Պարգևատրման չափը որոշվում է հետևյալ բանաձևով`

Պ= Ահիմ \* Պդ/100%,

որտեղ` Պդ -պարգևատրման դրույքաչափ, %

Պ = 444500\*25/100=111․125 դրամ:

Ընդամենը հիմնական աշխատավարձը կկազմի`

111․125 դրամ/ամսական կամ`

1․333․500 դրամ/տարեկան:

1. **Աշխատողների լրացուցիչ աշխատավարձի հաշվարկը**

Լրացուցիչ աշխատավարձի մեջ մտնում են`հերթական և լրացուցիչ գործողումների, արձակուրդների վճարները, պետական հանձնարարականների կատարման հետ կապված ծախսերը և այլն: Աշխատողների լրացուցիչ աշխատավարձը հաշվարկվում է հետևյալ բանաձևով`

Ա լր. = ԸԱհիմ. \* Ալր.դ/100,

որտեղ` ԸԱլր.դ – Ընդհանուր հիմնական աշխատավարձն է, իսկ Ալր.դ-ն լրացուցիչ աշխատավարձի դրույքաչափն է,%:

Ալր. = 111125 \*18/100 = 20002դրամ/ամսական կամ

240․030դրամ/տարեկան:

**5․ Սարքավորումների պահպանման և շահագործման ծախսերի հաշվարկը**

Սարքավորումների պահպանման և շահագործման ծախսերի թվին են պատկանում ամորտիզացիոն, ընթացիկ վերանորոգման, տրանզիտորային միջոցների, գործիքների և հարմարանքների վերանորոգման և այլ ծախսերը: Սարքավորումների պահպանման և շահագործման ծախսերի թվին են պատկանում ամորտիզացիոն, ընթացիկ վերանորոգման, տրանզիստորային միջոցների, գործիքների և հարմարանքների վերանորոգման և այլ ծախսերը:

**5.1 Հիմնական միջոցների ամորտիզացիա**

Հիմնական միջոցների տարեկան ամորտիզացիան (ԱՏ) հաշվարկվում է հետևյալ բանաձևով`

**ԱՏ=ՀԱ/Ն,**

որտեղ ՀԱ-ն հիմնական միջոցների սկզբնական արժեքն է, Ն-ն հիմնական միջոցների օգտակար գործունեության ժամկետն է:

*Աղյուսակ 3.*

|  |  |  |  |
| --- | --- | --- | --- |
| Հիմնական միջոցի անվանումն ու տեսակը | Հիմնական միջոցի սկզբնական արժեքը | Ամորտիզացիոն հատկացումներ | |
| ՀՄ օգտակար գործողության ժամկետ, տարի | Ամորտիզացիոն ծախս |
| Սարքավորումներ | Համակարգիչ | 5 | 300 000 |
| Ընդամենը |  |  | 300 000 |

Սարքավորումների պահպանման և շահագործման ծախսերի գումարը որոշվում է հետևյալ կերպ`

Ծպ.շ. = Ահիմ\*12\* Ծպ.շ.դ./100,

որտեղ` Ծպ.շ.դ. - սարքավորումների պահպանման և շահագործման ծախսերի դրույքաչափն է, Ահիմ.–աշխատողների տարեկան հիմնական աշխատավարձը:

Ծպ.շ. = 1333500\*12\*0․9/100 = 144018 դրամ:

Սարքավորումների ընթացիկ վերանորոգման ծախսը որոշվում է հետևյալ բանաձևով`

Ծը.վ. = Ահիմ \*12\* Ծը.վ.դ./100,

որտեղ` Ծը.վ.դ. - սարքավորումների ընթացիք վերանորոգման դրույքաչափն է:

Ծը.վ. = 1333500\*12\*6.1/100 = 976122 դրամ:

*Աղյուսակ 4.*

|  |  |  |
| --- | --- | --- |
| Ծախսի անվանումը | Դրույքաչափը | Տարեկան ծախսը |
| Սարքավորումների պահպանման և շահագործման ծախսեր | 0․9 | 144 018 |
| Սարքավորումների ընթացիկ վերանորոգում | 6.1 | 976 122 |
| Սարքավորումների ամորտիզացիա | - | 300 000 |
| Ընդամենը | | 1,420,140 |

**11**

**6. Տարածքի վարձակալության ծախսի հաշվարկը**

Գործունեության արդյունավետության հիմնական պայմաններից մեկը տարածքի ճիշտ ընտրությունն է: Վարձակալվող տարածքը կգտնվի Երևանի կենտրոնական մասում, Կոմիտաս 54Վ հասցեում․Այս տարածքում անհրաժեշտ տարածքի վարձակալության ամսեկան ծախսը կկազմի 200000դրամ կամ 200000\*12=2400000դրամ տարեկան։

**7․ Ընդհանուր տնտեսական ծախսերի հաշվարկը**

Ընդհանուր տնտեսավարման ծախսերի մեջ մտում են ձեռնարկության ընդհանուր կառավարման-ադմինիստրատիվ` գործարանը կառավարող անձնակազմի աշխաավարձի, գործուղման, տպագրական, փոստային-հեռագրային ծախսերը և այլ ծախսեր: Այն որոշվում է հետևյալ բանաձևով`

ԸՏԾ = Ա հիմ\* %ԸՍԾ/100,

որտեղ` %ԸՏԾ - Ընդհանուր տնտեսավարման ծախսերի տոկոսն է %:

ԸՏԾ = 1333500\*117/100=1560195 դրամ:

**8. Փաթեթի ընդհանուր ինքնարժքի կալկուլացիան**

Լրիվ Ինքնարժեքի կալկուլացիան բերված է աղյուսակ 5-ում:

*Աղյուսակ 5.*

|  |  |  |
| --- | --- | --- |
| N | Ծախսերի հոդվածի անվանումը | գումարը, դրամ |
| 1. | Համալրող առարկաներ | 1.940.000 |
| 2. | Էլեկտրաէներգիա | 120 000 |
| 3. | Ընդհանուր հիմնական աշխատավարձ | 1333500 |
| 4. | Լրացուցիչ աշխատավարձ | 240․030 |
| 5. | Սարքավորումների պահպանման և  շահագործման ծախսեր | 1,420,140 |
| 6. | Տարածքի վարձակալություն | 2400000 |
| 7․ | Ընդհանուր տնտեսական ծախսեր | 1560195 |
|  | **Լրիվ ինքնարժեք** | 7,267,865 |

Ուսումնական պլանի ձևավորման ավտոմատացված համակարգի մշակման ծրագրային փաթեթի լրիվ ինքնարժեքը կկազմի `

Իլրիվ= 7,267,865 դրամ:

**9. Շահույթի և միավորի գնի հաշվարկը**

Փաթեթի գինն իր մեջ ընդգրկում է ընկերության շահույթն ու լրիվ ինքնարժեքը: Շահույթի հաշվարկն իրականացվում է հետևյալ բանաձևով`

Շ = Ի լրիվ\*%Շ/100,

որտեղ` Իլրիվ – փաթեթի լրիվ ինքնարժեքն է, %Շ –շահույթի դրույքաչափը, %

Շ = 7,267,865\*22/100=1598930.3 դրամ.

Գնի հաշվարկը իրականացվում է հետևյալ կերպ`

Գ= Իլրիվ + Շ;

Գ = 7267865+1598930.3 = 8.866.795 դրամ:

Փաթեթի բացթողնման գինը որոշվում է հետևյալ բանաձևով`

Գբաց. = Գ + ԱԱՀ,

որտեղ` Գբաց.– փաթեթի բացթողնման գինն է, ԱԱՀ-ն ավելացված արժեքի հարկը (20%):

Գբաց. = 8.866.795+8.866.795\*22/100=3901389.8դրամ:

Այսպիսով, կատարված հաշվարկների արդյունքում ստացանք, որ համակարգչային հավելված փաթեթի լրիվ ինքնարժեքը կկազմի 7,267,865դրամ, իսկ գինը`3,901,389.8 դրամ:

**Գլուխ 4**

**Բնապահպանություն**

**Պլաստիկ թափոնների վերամշակման եղանակները**

Թափոնները մարդու կենցաղային և արդյունաբերական գործունեության ժամանակ նյութերի ու էներգիայի փոխակերպման հետևանքով առաջացող արգասիքներ են, որոնք չունեն հետագա օգտագործելի հատկություններ։ Թափոնները մթնոլորտային օդի,  ընդերքի,  ջրերի  և  հողերի աղտոտման աղբյուր են։

Տարբերում են

* կենցաղային,
* արդյունաբերական
* խիստ վտանգավոր թափոններ։

Կենցաղային (սպառման) և  արդյունաբերական (արտադրական) թափոնները սպառման ու արտադրության ընթացքում գոյացած հումքի, նյութերի և այլ արգասիքների, արտադրանքի կամ մթերքի մնացորդներ են, ինչպես նաև ապրանքներ (արտադրանք), որոնք չեն համապատասխանում սահմանված պահանջներին (խոտան) կամ կորցրել են իրենց սկզբնական սպառողական հատկությունները։ Արդյունաբերական թափոններ են նաև օգտակար հանածոների արդյունահանման ապարները, անտառհատման մնացորդները։

Վտանգավոր թափոններն իրենց ֆիզիկական, քիմիական կամ կենսաբանական հատկություններով կարող են վտանգ ստեղծել մարդու առողջության ու շրջակա միջավայրի համար, ուստի և պահանջում են իրենց հետ վարվելու հատուկ եղանակներ ու միջոցներ։ Վտանգավոր թափոններ են նաև այն արտադրանքը, հումքը և նյութերը, որոնք չեն համապատասխանում սահմանված չափորոշիչներին և, էկոլոգիական ու սանիտարահիգիենային պահանջներից ելնելով, պիտանի չեն օգտագործման։ Վտանգավոր թափոնները դասակարգվում են ՀՀ բնապահպանության նախարարի «ՀՀ տարածքում առաջացող թափոնների ըստ վտանգավորության աստիճանի դասակարգման մասին» հրահանգի համաձայն։

Ըստ ագրեգատային վիճակի՝ թափոնները լինում են

* պինդ,
* հեղուկ
* գազային։

Պինդ թափոններ են մետաղների, պլաստմասսաների, ապակու, թղթի մնացորդները, աղբը (սննդի մնացորդները), մաշված մեքենաները, սարքավորումները, գործիքները և այլն։

Հեղուկ թափոններ են արդյունաբերական և կենցաղային կեղտաջրերը։

Գազային թափոններ են ձեռնարկությունների և  տրանսպորտի արտանետումները։

Թափոնների գործածությունը գործողությունների համախումբ է, որն ուղղված է դրանց գոյացման կանխարգելմանը, հավաքմանը, փոխադրմանը, պահմանը, մշակմանը, վերամշակմանը, օգտահանմանը, հեռացմանը, վնասազերծմանը և թաղմանը։

Թափոնների հավաքումը կապված է դրանց հեռացման և հատուկ հատկացված տեղերում կամ օբյեկտներում տեղադրման հետ (ներառյալ նաև տեսակավորումը)։

Պահումը թափոնների ժամանակավոր տեղադրումն է հատուկ հատկացված տեղերում՝ հետագա օգտահանման կամ հեռացման նպատակով։

Օգտագործումը թափոնների կիրառումն է ապրանքների արտադրության, էներգիայի  ստացման կամ այլ նպատակներով։

Մշակումը կամ վերամշակումը տեխնոլոգիական շարժընթացների իրականացումն է, որոնցով փոխվում են թափոնների ֆիզիկական, քիմիական կամ կենսաբանական հատկությունները (օրինակ՝ արտադրական թափոնից  ստացվել է բարձրարժեք էթինոլային լաք)։

Օգտահանումը թափոնների օգտագործումն է որպես երկրորդային նյութ (օրինակ՝ տուֆային թափոնները, որպես ջրախառնիչ հավելույթներ, մեծացնում են արտադրանքի ամրությունը, «Կոտայք» գարեջրի գործարանի արտադրական թափոններն օգտագործվում են որպես անասնակեր) կամ էներգետիկ պաշար։

Թափոնների հեռացումը գործողությունների համախումբ է, որն օգտահանման նպատակ չունի։

Վնասազերծումը թափոնների վտանգավոր հատկությունների նվազեցումը կամ վերացումն է՝ մեխանիկական, ֆիզիկաքիմիական կամ կենսաբանական մշակմամբ։

Տեղադրումը թափոնների մեկուսացումն է, հետագա օգտագործման բացառումը և ուղղված է դրանց չեզոքացմանն ու շրջակա միջավայրում վտանգավոր նյութերի հետագա տարածման կանխմանը։

Թաղումը թափոնների վերջնական տեղադրումն է հատուկ առանձնացված տեղերում, որով բացառվում է դրանց վտանգավոր ներգործությունը մարդու առողջության և շրջակա միջավայրի վրա։

Փոխադրումը թափոնների տեղափոխումն է դրանց գոյացման կամ պահման վայրերից դեպի մշակման, օգտահանման կամ հեռացման վայրեր։

Խոսենք պլաստիկ թափոնների վերամշակման փուլերից և եղանակներից։ Պլաստիկ թափոնները դասակարգվում են որպես պինդ թափոններ։ Ներկայումս արտադրության մեջ գնալով մեծանում է պլաստիկ նյութերի օգտագործումը։ Այդ նյութերը օգտագործվում են շշերի, փաթեթների, ցանցերի, խողովակների և շատ այլ տեսակի ապրանքներում։

Ինչպես հայտնի է, ներկայումս համակարգչի գրեթե 80%-ը պլաստիկ նյութեր են։ Երբ համակարգիչը դառնում է օգտագործելու համար ոչ պիտանի, առաջանում է պլաստիկ թափոն, որը անհրաժեշտ է վերամշակել և օգտագործել որպես հումք հետագա արտադրության մեջ։ Բացի այդ, համակարգչային տեխնիկայի կազմի մեջ են մտնում նաև վնասակար քիմիական նյութեր։ Օրինակ՝ համակարգչի մոնիտորը և մի շարք այլ մասեր պարունակում են այնպիսի վնասակար նյութեր, ինչպիսիք են սնդիկը, կապարը, կադմիումը և այլն։ Հետևաբար, այս նյութերի հետ ոչ ճիշտ կերպով վարվելու դեպքում շրջակա միջավայրին կարող է հասցվել մեծ վնաս։

Ներկայումս գնալով մեծանում է պլաստիկ թափոնների քանակը, որոնք ոչ միայն դառնում են բեռ, այլև աղտոտում են շրջակա միջավայրը։ Պլաստիկ նյութերը պատկանում են այն նյութերի շարքին, որոնք ժամանակի ընթացքում չեն քայքայվում, իսկ այրվելուց արձակում են այնպիսի թունավոր նյութեր, որոնք օրգանիզմից դուրս բերելն անհնար է։ Հետևաբար՝ պլաստիկ արտադրանքը անհրաժեշտ է վերամշակել։

Մեր օրերում պոլիմերային նյութերի վերամշակման խնդիրը լուրջ բնույթ է կրում ոչ միայն շրջակա միջավայրի պահպանության, այլ նաև պոլիմերային հումքի պակասի պատճառով։ 1 կգ պլաստիկ թափոնից վերամշակման արդյունքում կարելի է ստանալ 0,8 կգ երկրորդային հումք։

Պլաստիկ նյութերի տարածվածությունը արտադրության մեջ բացատրվում է նրա հեշտ ստացմամբ և օգտագործման հարմարավետությունով։ Օրինակ՝ պլաստիկ նյութը արտադրությունում որպես պատրաստի մաս ստանալու համար ծախսվում է 21%-ով պակաս էներգիա, քան ապակի ստանալուց։

Պլաստիկ թափոնների վերամշակումը բաղկացած է մի քանի փուլերից․

* թափոնների հավաքում,
* տեսակավորում (ըստ գույնի, որակի, մաքուր/կեղտոտ թափոնների),
* սեղմում
* վերամշակում (կտրում, լվացում, չորացում, արտաքին թաղանթի մշակում),
* նոր արտադրանքի արտադրություն։

Գոյություն ունեն պլաստիկ թափոնների վերամշակման մի քանի մեթոդներ․

1. Պիրոլիզ – նյութերի ջերմային տարրալուծում թթվածնի առկայությամբ կամ առանց դրա։
2. Հիդրոլիզ – տեղի է ունենում բարձր ջերմաստիճանի և ճնշման պայմաններում։ Այս եղանակը ավելի շահավետ է, քան պիրոլիզը, քանի որ շրջանառության մեջ են վերադարձվում բարձրորակ քիմիական արտադրանքներ։
3. Գլիկոլիզ – մշակումը տեղի է ունենում մեծ ճնշման և ջերմաստիճանի պայմաններում էթիլենգլիկոլի և կատալիզատորի առկայությամբ՝ մինչև մաքուր արտադրանքի ստացումը։ Այս մեթոդը ավելի շահավետ է, քան հիդրոլիզը։
4. Այնուամենայնիվ, պլաստիկ թափոնների վերամշակման ամենատարածված ջերմային մեթոդը մեթանոլիզն է՝ թափոնների առանձնացումը մեթանոլի օգնությամբ։

## Գլուխ 5

**Կենսագործունեության անվտանգություն**

## Համակարգչի ազդեցությունը մարդու օրգանիզմի վրա

Համակարգչի կիրառությունը լայն տարածում ունի ոչ միայն արտադրամասերում և գիտական լաբորատորիաներում, այլ նաև ուսանողական լսարաններում և դպրոցական դասարաններում:Անընդհատ աճում է այն մասնագետների թիվը,որոնք աշխատում են անհատական համակարգչով, որն էլ դառնում է նրանց աշխատանքային հիմնական գործիքը: Թե’ էկոնոմիկական, թե’ գիտական ձեռքբերումներն այժմ անհնար են առանց արագ և ճշգրիտ ինֆորմացիոն կապի և առանց հատուկ ուսուցում ստացած անհատների:Մինչդեռ համակարգչի էկրանի առաջ երկար ժամանակ մնալը առանց անհրաժեշտ կանոնները պահելու վնասակար է օպերատորի առողջության համար: Առաջին հերթին նրանք հայտնաբերում են տեսողության խախտում, ձեռքի մկանների և ողնաշարի հոգնածություն, ընդհանուր թուլություն:Օրգանիզմի վրա համակարգչի վնասակար ազդեցության հիմնական գործոններն են էլեկտրամագնիսական դաշտը և ճառագայթումը, նկարի էլեկտրոնային տեղափոխումը և նրա թարթումը էկրանի վրա:Տեսամոնիտորները տեսաինֆորմացիայի արտահայտման միջոցներ են, որոնք ապահովում են մարդու և էլեկտրոնային-հաշվողական մեքենայի միջև էֆեկտիվ ինֆորմացիոն փոխազդեցություն, տարբեր ավտոմատացված կառավարման համակարգերում ստանում են լայն տարածում և հանդիսանում են ինֆորմացիայի փոխանակման միջոց միկրո և մակրոէլեկտրոնային հաշվողական մեքենաներում:

Համակարգիչները լինում են ոչ միայն գիտական լաբորատրորիաներում և արտադրամասերում, այլ տեղադրվում են նաև տներում: Սա մարդկանց համար ստեղծում է հարմարավետություն և հեշտացնում գործնական կյանքը:

Ամեն օր օպերատորները մի քանի ժամ անցկացնում են էլեկտրոնային տեսամոնիտորների էկրանների առաջ, որը սանիտարահիգիենիկ նորմերի և կանոնների անտեսման դեպքում, կարող է հանգեցնել որոշ մասնագիտական հիվանդությունների: Օպերատորների առողջական վիճակի վրա կարող են ազդել այնպիսի վնասակար գործոններ, ինչպիսիք են մարմնի երկարատև անփոփոխ դիրքը, որն առաջացնում է մկանա-կմախքային խախտումներ, աչքերի անընդհատ լարվածությունը, ռադիացիայի ազդեցությունը, էլեկտրաստատիկ և էլեկտրամագնիսական դաշտերի ազդեցությունը, որը կարող է առաջացնել մաշկային հիվանդություններ, գլխացավեր և մի շարք օրգանների ֆունկցիաների խախտումներ:   
 Վերջին ժամանակներս ուշադրություն է գրավում այնպիսի մի ազդեցություն, ինչպիսին տեխնոսթրեսն է: Բանն այն է, որ տեսամոնիտորների ներդրումից հետո նրանց հետ աշխատողները փորձում են ստանալ հետևյալ հարցերի պատասխանները. տեսամոնիտորների ուժային տրանսֆորմատորը վտանգավո՞ր է արդյոք ճառագայթման տեսանկյունից, տեսամոնիտորները առաջացնո՞ւմ են արդյոք վտանգավոր այլ ճառագայթումներ, հնարավո՞ր են արդյոք առողջության հետ կապված այլ խնդիրներ: Այս հարցերին անգամ բացասական պատասխանների դեպքում որոշակի անորոշություն է մնում, որը կարող է առաջացնել ինքնազգացողության վատացում և նույնիսկ դիսփլեյից վախենալու զարգացում: Այդ պատճառով տեսամոնիտորի էկրանի առաջ գտնվող մարդու հոգեկան վիճակը կարևորագույն խնդիրներից մեկն է, որի լուծման համար աշխատում են հազարավոր մասնագետներ:Աշխատանքային պայմանների և աշխատողների առողջության փոխադարձ կապի ազդեցությունը պարունակում էբժշկական հետազոտություն,աշխատանքային խնդիրների վերլուծություն, մտավոր ծանրաբեռնվածության մակարդակը և ծանրաբեռնվածությունը էրիթելային ապարատի վրա,այն ժամանակի քանակական գնահատականը, որն անհրաժեշտ է տվյալ խնդիրների լուծման համար,հիգիենիկ պայմանների հետազոտություն-օդում որակական պարամետրերի փոփոխությունը,աշխատանքի ճշտության և օդափոխող համակարգերի էֆեկտիվության ստուգում, շրջակա աղմուկի վերլուծություն,լուսատեխնիկական պայմանների վերլուծություն:Իոնացում չառաջացնող էլեկտրամագնիսական ճառագայթումը ոչ օպտիկական հաճախության միջակայքում կարող է վնասել առողջությանը, նշանակություն ունեն նաև դաշտի լարվածությունը, հաճախականության միջակայքը, ճառագայթման ձևը (իմպուլսային կամ անընդհատ) և ազդեցեցության ժամանակը: Որոշ աշխատավայրերում տեսամոնիտորները հանդիսանում են ոչ իոնացնող էլեկտրամագնիսական, օպտիկական և սուբօպտիկական ճառագայթման ուժեղ աղբյուրներ: Աշխատանքային ծանրաբեռնվածությունը կարող է հանգեցնել մաշկային հիվանդությունների որոշ ախտանիշների, սակայն չի կարելի համարձակ հաստատել , որ տեսամոնիտորների ազդեցության արդյունք է:Տեսողական ապարատի ֆունկցիայի խանգարումը կարելի է բացատրել հետևյալ գործոններով`համակարգչի էկրանի պայծառության և շրջակա տարածքի լուսավորության միջև կտրուկ կոնտրաստով (նախընտրելի է միջին կոնտրաստը)աշխատավայրի վատ լուսավորությամբԷկրանի և նկարի թարթումը և ցնցումը, կոնտրաստի կտրուկ անկումը արտաքին լուսավորության պայմաններում ի հայտ է գալիս տեսամոնիտորների աշխատանքի ժամանակ:Այլընտրանքային տեխնոլոգիաներ են հանդիսանում նորագույն հեղուկ-բյուրեղա տեսամոնիտորները: Պլազմային տեխնոլոգիան պրակտիկորեն բացառում է թարթումը: Տնտեսապես ավելի հարմար են հեղուկ-բյուրեղյա տեսամոնիտորները` ինչպես շարժական, այնպես էլ ստացիոնար համակարգիչների համար:

Էլեկտրոնաճառագայթային խողովակներով տեսամոնիտորները հանդիսանում են թույլ ռենտգենյան, ուլտրամանուշակագույն, ինֆրակարմիր, տեսանելի, ռադիոհաճախականային, բարձր և ցածրահաճախականային, էլեկտրամագնիսական ճառագայթման պոտենցիալ աղբյուրներ:

Էլեկտրամագնիսական ճառագայթման ռադիոհաճախականության միջակայքերի աղբյուրները կարող են լինել էլեկտրոնաճառագայթային խողովակի ճառագայթի հորիզոնական շեղումը:Լույսը հանդիսանում է մարդու գոյության կարևորագույն պայմաններից մեկը: Այն ազդում է վիճակի վրա: Ճիշտ կազմակերպված լուսավորությունը կարգավորում է նյարդային համակարգի գործունեության և բարձրացնում աշխատունակությունը: Ավելի քիչ լուսավորության պայմաններում մարդն աշխատում է ավելի քիչ արդյունավետությամբ,շուտ հոգնում է, սխալ գործողությունների հավանականությունն աճում է, որը կարող է հանգեցնել վնասվածքների: Լույսի սպեկտրալ կազմն ազդում է աշխատանքի արդյունավետության վրա: Աշխատավայրի լուսավորությունը պետք է կազմակերպված լինի ըստ հետևյալ պայմանների:աշխատանքային միջավայրի լուսավորության մակարդակը պետք է համապատասխանի տվյալ աշխատանքի համար նախատեսված հիգիենիկ նորմերին,պետք է ապահովված լինի շրջակա տարածքի լուսավորության հավասարաչափ և հաստատուն մակարդակ, աշխատանքային միջավայրի լուսավորության և շրջակա տարածքի միջև կտրուկ կոնտրաստների բացակայությունը,արհեստական լույսը, որն օգտագործվում է աշխատավայրում, իր սպեկտրալ կազմով պետք է մոտ լինի բնական լույսին:Լուսավորությունը կարող է լինել բնական, արհեստական և խառը:Բնական լույսն իրականացվում է ապակու միջոցով: Արհեստական լուսավորությունը ֆունկցիոնալ նշանակության տեսանկյունից լինում են աշխատանքային , հերթապահ, վթարային, էվակուացիոն և պաշտպանող:Լույսի աղբյուրի դասավորությունից կախված արհեստական լուսավորությունը լինում է ընդհանուր, տեղային և կոմբինացված:

Մարդու վրա զգալի ազդեցություն է ցուցաբերում նաև աշխատավայրի միկրոկլիման: Միկրոկլիմայի առանձին պարամետրերի հանձնարարված նշանակությունից շեղումը իջեցնում է աշխատունակության մակարդակը, վատացնում աշխատողի ինքնազգացողությունը և կարող է հանգեցնել մասնագիտական հիվանդությունների:

Աշխատավայրում միկրոկլիմայի ձևավորման համար գլխավոր դեր են կատարում նաև ջերմաստիճանը, խոնավությունը և օդի շարժման արագությունը:Անկախ շրջակայքի միկրոկլիմայից մեծ նշանակություն ունեն օդում պարունակող խառնուրդները: Կախված այն ուժերից,որոնք պայմանավորված են օդի տեղաշարժով, տարբերում են բնական և արհեստական օդափոխություն:

**Գրականություն ցանկ**

* *Баричев С.Г., Гончаров В.В., Серов Р.Е.* 2.4.2. Стандарт AES. Алгоритм Rijdael // Основы современной криптографии. — М.: Горячая линия — Телеком, 2002. — С. 30—35. — (Специальность. Для высших учебных заведений). — 3000 экз. — [ISBN 5-93517-075-2](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D0%B6%D0%B5%D0%B1%D0%BD%D0%B0%D1%8F:%D0%98%D1%81%D1%82%D0%BE%D1%87%D0%BD%D0%B8%D0%BA%D0%B8_%D0%BA%D0%BD%D0%B8%D0%B3/5935170752)
* ***Шнайер Б.*** [Прикладная криптография. Протоколы, алгоритмы, исходные тексты на языке Си](http://www.ssl.stu.neva.ru/psw/crypto/appl_rus/appl_cryp.htm) = Applied Cryptography. Protocols, Algorithms and Source Code in C. — М.: [Триумф](http://ru.wikipedia.org/w/index.php?title=%D0%A2%D1%80%D0%B8%D1%83%D0%BC%D1%84_(%D0%B8%D0%B7%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D1%82%D0%B2%D0%BE)&action=edit&redlink=1), 2002. — 816 с. — 3000 экз. — [ISBN 5-89392-055-4](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D0%B6%D0%B5%D0%B1%D0%BD%D0%B0%D1%8F:%D0%98%D1%81%D1%82%D0%BE%D1%87%D0%BD%D0%B8%D0%BA%D0%B8_%D0%BA%D0%BD%D0%B8%D0%B3/5893920554)
* *Мэтью Мак-Дональд* WPF: Windows Presentation Foundation в .NET 3.5 с примерами на C# 2008 для профессионалов = Pro WPF in C# 2008: Windows Presentation Foundation with .NET 3.5. — 2-ое. — М.: [«Вильямс»](http://ru.wikipedia.org/w/index.php?title=%D0%92%D0%B8%D0%BB%D1%8C%D1%8F%D0%BC%D1%81_(%D0%B8%D0%B7%D0%B4%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D1%81%D1%82%D0%B2%D0%BE)&action=edit&redlink=1), 2008. — С. 25. — 928 с. — [ISBN 978-5-8459-1429-3](http://ru.wikipedia.org/wiki/%D0%A1%D0%BB%D1%83%D0%B6%D0%B5%D0%B1%D0%BD%D0%B0%D1%8F:%D0%98%D1%81%D1%82%D0%BE%D1%87%D0%BD%D0%B8%D0%BA%D0%B8_%D0%BA%D0%BD%D0%B8%D0%B3/9785845914293)
* Andrew Troelsen. Pro C# 5.0 and the .NET 4.5 Framework
* SQL в примерах и задачах; Учеб. пособие / И.Ф. Астахова, А.П. Тол-стобров,В.М. Мельников.— Мн.: Новое знание, 2002. — 176 с.
* Шилдт, Герберт. Ш57 С# 4.0: полное руководство. : Пер. с англ. — М. : ООО "И.Д. Вильяме", 2011. — 1056 с.: ил. — Парал. тит. англ. ISBN 978-5-8459-1684-6 (рус.)