

УДК 378  
ББК 74.58  
Ә 82

М.Әуезов атындағы Оңтүстік Қазақстан мемлекеттік университетінде «ӘУЕЗОВ ОҚУЛАРЫ - 12: «Ғылым, білім және мәдениеттің инновациялық бағыттарын дамытудағы аймақтық университеттің рөлі» атты халықаралық ғылыми-тәжірибелік конференция - Шымкент: М. Әуезов ат. ОҚМУ, 2014ж.

2 Т. 295 б. Тілдері: казак, орыс, ағылшын.

Международная научно-практическая конференция «АУЭЗОВСКИЕ ЧТЕНИЯ - 12: «Роль регионального университета в развитии инновационных направлений науки, образования и культуры» - Шымкент: ЮКГУ им. М.Ауэзова, 2014 г.

2 Т. 295 с. Языки: казахский, русский, английский.

The international scientific-practical .conference «AUEZOV READINGS -12: The role of regional university in science, education and culture innovative directions development)) - Shymkent: M.Auezov SKSU, 2014 y.

2 V. 295 p. Languages: kazakh, russian, english.

## ISBN 978-9965-03-246-7

Бас редактор: Мырхалыков Ж.Ү. - М.Әуезов атындағы ОҚМУ ректоры, т.ғ.д., профессор, ҰҒА корреспондент мүшесі.

Редакциялық алқа мүшелері: Сатаев М.И. - төрағаның орынбасары, ҒЖ және ХБ жөніндегі проректор, т.ғ.д., профессор, ҰҒА корреспондент мүшесі; Горяинов К.К. - з.ғ.д., профессор, Ресей Федералды қызметінің жазасын орындау ҒЗИ. Ресей; Дырка Стефан •• э.ғ.д., профессор, экономика ғылымдарының докторы, Верхнесилез экономикалық университеті, Польша; Меор Мохаммед Фаред - ассоциациялық профессор, Путра университеті, Малайзия; Олден А. - академик, Лондон Батые университетінің есептеуші техника және технология мектебі, Ұлыбритания; Окуян М.Д. - профессор, Балыкесир университеті, Түркия; Линда Лоутон - PhD докторы, профессор, Робер Горден атындағы университеті, Ұлыбритания; Мохд Хасан Бен Селамат - PhD докторы, профессор, Малайзия университеті, Малайзия; Ивахненко А.П.- PhD докторы, директор, Мұнай зерттеу орталығы, Heriot-Watt университету Шотландия; Елизавета Ф. - PhD докторы, профессор, Басел университеті, Австрия; Милу Ж. - т.ғ.д., профессор, Лорейн университеті, Франция; Петров В. - доктор, профессор, Левон Католик университету Бельгия; Радюк С.Н. - PhD докторы, ассоциациялық профессор, Оңтүстік әдістемелік университеті, АҚШ; Славинская Н.А. - бас ғылыми қызметкері, неміс аэрОкосмостық агенттігі, Германия; Жонго Ок - PhD докторы, профессор, Сеул ұлттық техникалық университету Корея, Беккерман М. - профессор, Ньюрки университеті, АҚШ; Крючкова О.Ю. - |ф4^, профессор, Н.Г.Чернышевский атындағы Саратов мемлекеттік университету Ресей; Марфенин Н.Н. - б.ғ.д., профессор, М.В. Ломоносов атындағы Мәскеу мемлекеттік университеті, Ресей; Бишімбаев У.Қ. - т.ғ.д., профессор, ҚР ҰҒА академигі, Қазақстан; Жұрынов М.Ж - х.ғ.д., профессор, ҚР ҰҒА академигі, Қазақстан; Айменов Ж.Т. - т.ғ.д., профессор; ҚР ҰЖҒА академигі, М.Әуезов атындағы ОҚМУ, Қазақстан Байтанаев Б.А - т.ғ.д., профессор, ҚР ҰҒА корреспондент мүшесі, М.Әуезов атындағы ОҚМУ; Калменов Т.Ш. - ф-м.ғ.д., профессор, ҚР ҰҒА академигі, Қазақстан; Молдабеков Ш.М. - т.ғ.д., профессор, ҚР ҰИА, Қазақстан; Надиров Н.К. - х.ғ.д., профессор, ҚР ҰҒА академигі, Қазақстан; Есимов Б.О. - г-м.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Шакиров Б.С. - т.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Кулымбетова А.Е. - п.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Қалыбекова А.А. - п.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Мұсаева Н.Р. - филос.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Мырзахметов М. - ф.ғ.д., профессор, Қазақстан; Назарбекова С.П. - х.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Ташимов Л.Т. - т.ғ.д. профессор, ҚР ҰҒА корреспондент мүшесі, М.Әуезов атындағы ОҚМУ; Ниязбекова Р.К. - э.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Волненко А.А. - т.ғ.д., профессор, М.Әуезов атындағы ОҚМУ; Тлеулов Ә.М. - п.ғ.к., доцент, М.Әуезов атындағы ОҚМУ-

УДК 378

ББК 74.58

## ISBN 978-9965-03-246-7

М.Әуезов атындағы Оңтүстік Қазақстан мемлекеттік университету 2014  
Южно-Казахстанский государственный университет им.М.Ауэзова, 2014

**ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ БІЛІМ  
М.ӘУЕЗОВ атындағы Оңтүстік Қазақстан;**

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
ЮЖНО-КАЗАХСТАНСКИЙ ГОСУДАРСТВЕННЫЙ**

**MINISTRY OF EDUCATION AND SCIENCE  
M. AUEZOV SOUTH KAZAKHSTAN**

**ҒЫЛЫМ АЛҚАМУ**  
**"ІК УНИВЕРСИТЕТІ"**

**КАЗАХСТАН  
УНИВЕРСИТЕТІ**

**KHSTAN**



**«ӘУЕЗОВ ОҚУЛАРЫ - 12: «ҒЫЛЫМ, БІЛІМ ЖӘНЕ МӘДЕНИЕТТІҢ ИННОВАЦИЯЛЫҚ  
БАҒЫТТАРЫН ДАМУДАҒЫ АЙМАҚТЫҚ УНИВЕРСИТЕТТІҢ РӨЛІ»  
ХАЛЫҚАРАЛЫҚ ҒЫЛЫМИ-ТӘЖІРИБЕЛІК КОНФЕРЕНЦИЯНЫҢ**

## **ЕҢБЕКТЕРІ**

**ТРУДЫ**

**V**

**МЕЖДУНАРОДНОЙ НАУЧНО-ПРАКТИЧЕСКОЙ КОНФЕРЕНЦИИ  
«АУЕЗОВСКИЕ ЧТЕНИЯ - 12: «РОЛЬ РЕГИОНАЛЬНОГО УНИВЕРСИТЕТА  
В РАЗВИТИИ ИННОВАЦИОННЫХ НАПРАВЛЕНИЙ НАУКИ  
ОБРАЗОВАНИЯ И КУЛЬТУРЫ**

**WORKS**

**OF THE INTERNATIONAL SCIENTIFIC-PRACTICAL CONFERENCE  
«AUEZOV READINGS - 12: «THE ROLE OF REGIONAL UNIVERSITY  
IN SCIENCE, EDUCATION AND CULTURE INNOVATIVE  
DIRECTIONS DEVELOPMENT\***

**Шымкент 2014**

Внедрение компьютерных технологий в процесс обучения истории создаёт предпосылки для интенсификации образовательного процесса. Использование компьютерных технологий вносит изменения в цели и содержание обучения, появляются новые методы и организационные формы обучения. КТ позволяют использовать на практике психолого-педагогические разработки, обеспечивающие переход от механического усвоения знаний к овладению умением самостоятельно приобретать новые знания, развивать информационную компетентность будущего специалиста.

Очевидно, что рассмотренная информационная технология позволяет преподавать не только географию, но и историю на более высоком научно-теоретическом и методическом уровнях, интегрировать знания по предмету, а обучаемым ощущать себя активными участниками процесса обучения, получать новые знания, овладевать соответствующими компетенциями.

#### Литература

1. Веселовский А.В. ГИС-технологии и проблемы геоинформатики. Географические информационные системы научного центра «Минерал» // Вестник ОГГГН РАН, 1999. № 1(7). С. 54-61.
2. Макарова Л.Н. Применение технических средств на уроках географии // Вопросы Интернет образования, 2006. № 36
3. Хасаншина Н.З. Геоинформационные технологии как средство интеграции знаний по информатике и географии // XII международная конференция-выставка «Информационные технологии в образовании» («ИТО-2002»)
4. Грибан О.Н. Новые информационные технологии в процессе обучения истории: анализ образовательной практики // Историческая наука и историческое образование как факторы гуманизации общества: сб. науч. ст. / УрГПУ. - Екатеринбург, 2012. Ч. I. - С. 192-199.

ЭОЖ(378.016.02:004.032.6:574)

#### С# ПРОГРАММАЛАУ ТІЛШІ ЕРЕКШЕЛІКТЕРІ ЖӘНЕ ҒЫЛЫМИ ЕСЕПТЕУ ЖҰМЫСТАРЫНДА ҚОЛДАНУ

Мирзахмедова Г.А., Самбетбаева А.К.

Эл-Фараби атындағы Қазак Ұлттық университеті, Алматы, Қазақстан

#### Резюме

*Язык С# довольно успешно использовался в проектах многих типов, в том числе для разработки Web-приложений, баз данных, GUI и т. д. Одним из последних рубежей применения С# кода вполне могут стать научные вычисления. Язык С# заслужил большое уважение и популярность среди разработчиков самых разных программных продуктов. Последнюю пару лет С# играл важную роль в производстве устойчивых к сбоям продуктов - от настольных приложений до Web сервисов, от высокоуровневых решений в автоматизации бизнес - процессов до программ системного уровня и от однопользовательских продуктов до корпоративных решений в сетевых распределенных средах.*

#### Summary

*C # language quite successfully used in many types of projects, including for the development of Web-based applications , database, GUI , etc. One of the last frontiers of application C # code may well be scientific computing. C # language has earned great respect and popularity among developers a variety of software products. The last couple of years, C # has played an important role in producing robust products from desktop applications to Web services, from high automation of business processes to software and system-level products from single user to enterprise solutions in network distributed environments .*

С# программалау тілі көптеген типтегі, атап айтар болсақ Web-қосымшаларын, мәліметтер қоры, GUI және т.с.с жобаларды құру барысында кең қолданылады. Ғылыми және математикалық жобаларды құру барысында С# программалау тілін FORTRAN және С++ программалау тілдерімен салыстыруға бола ма? ЛТ жүктеушінің, Microsoft (Microsoft intermediate language, MSIL) аралық тілінің және қалдықтар жиындығының (garbage collector) программаның енімділігіне қалай әсер ететінін анықтау үшін NET программалау тілдеріндегі жалпы тілдік қолданушы ортаны (common language runtime, CLR) зерттейміз.

Бұл жұмыста біз С# программалау тілінің өңделу жылдамдығы қиынға соғатын программалар кодтарды жеңіл құруға мүмкіндік беретін кейбір іпгі ерекшеліктерін қарастырамыз. С# программалау тілінің ғылыми жобаларды құру, сандық есептеулер жүргізу барысында маңызы зор екеніне көз жеткізуге болады. Сонымен қатар жадыны басқару кезінде жүктелу жылдамдығы баяу орындалатын жобалардың жеңіл және жылдам орындалатынына көз жеткізуге болады; программалық код есептеу амалдарын (discard)

жүргізу барысында жадыда үлкен орынды алмайтын болғандықтан калдықтар жиынтығында кідіріске рпырамайды.

Ғылыми жұмыстарды жобалау барысында программалау тілдерінің негізгі ерекшелігі оның өнімділігі болып табылады. Кодтарды жүктеуіштер мен генераторлар өнімділікті шектеуші факторлар болып табылады. Мысалы, кең таралған C++ программалау тілінің жүктеуіштері кодтарды генерациялау мен тиімділеу жұмыстарын жақсы атқарады. Айтар болсақ, C++ тілінде уақыттық нысандардың санының аса жоғары болмағаны жөн. Сондықтан да математикалық өрнектерді дәл шепгуге мүмкіндік беретін өрнектер нұсқаушының қолданған жөн. Нәтижесінде математикалық амалдардың орындау барысындағы көптеген шателіктерге жол берілмейді.

Өзге де .NET объектіге бағытталған тілдер секілді C# тілі жалпы программалау тілдерінің орындалу ортасында (CLR) жүзеге асырылатын MSIL (Microsoft intermediate language)-ге жүктеледі. CLR қысқаша тиімділеуші JIT жүктеуішінің және калдықтар жиынтығының кешені түрінде көрсетуге болады. C# тілі CLR функционалдылығының басым бөлігін қолданады және көрсетеді, сондықтан да орындалушы ортада қандай шал орындалып жатқанын бөліктеп қарастырған маңызды.

Ғалымдардың кілттік талаптарының бірі - кодтардың портталуы (платформалар арасында алмастыруға болады). Ғылыми - зерттеу институттары мен зертханалары жоғары деңгейлі платформалар мен машиналармен жұмыс істейді. Ондай ұжымдарда программалық код дәл нәтижені табу үшін әртүрлі машиналарда жүктелетін. Бірақ нақты жауапты алу аса оңайға соқпайтын. Мысалы, көптеген күрделі масштабты жобалар бірнеше тілде жазылған программалық кодтарды қолдану арқылы өңделеді; сондықтан бір платформада орындалған программалық кодтың келесі платформада жұмыс жасайтынына ешкім кепілдік бере алмайды.

CLR кітапханалар мен қосымшаларды MSIL-де жинақталған бірнеше программалау тілінде жазуға мүмкіндік береді. Сонан соң MSIL кез келген өзіне сәйкес келетін архитектурада жүзеге асырыла береді. Енді ғалымдар өздерінің математикалық кітапханаларын FORTRAN тілінде жазып оны C++ сәйкестендіреді және нәтижелерін ғаламторда жариялау үшін C# программалау тілін және ASP.NET қолдана алады.

Java Virtual Machine (JVM) мен салыстырғанда CLR - бұл көптеген программалық тілдерге арналған орта және мақсатты платформа. Сонымен қатар CLR мәліметтер арасындағы өзара байланысты, программалық тілдердің арасындағы өнімділіктерін ажыратуды қамтамасыз етеді. MSIL-кодта жүктелетін бірнеше программалық тілдер бар. Мұндай тілдердің құрамына Ada, C, C++, Caml, COBOL, Eiffel, FORTRAN, Java, LISP, Logo, Mixal, Pascal, Perl, PHP, Python, Scheme и Smalltalk кіреді.

JIT жүктемесі - бірнеше программаларды тиімді пайдалануға мүмкіндік беретін технологияның аса ерекше бөлігі. JIT жүктеуіші теориялық түрде бірнеше рет қайта жүктелетін программалық код аса тиімді генерациялау арқылы жинақталған ақпараттарды қолдануға мүмкіндік береді. Қазір жүктеуіш машиналық кодты әрбір әдіс үшін бір рет қана генерациялайды. Генерациядан соң машиналық код компьютердің жылдамдығына байланысты орындалады.

Ғылыми программалау жұмыстарында бұл ыңғайлы болуы мүмкін. Ғылыми есептеулер коды негізінде сандық амалдарды орындайды. Мұндай сандық амалдарды аз ғана уақытта орындау үшін біз аппаратты жабдықтарды тиімді пайдалана білуіміз керек.

SciMark эталонды тесттері ғылыми қосымшаларда кең таралған есептеу амалдарын пішіндейтін бірнеше өзектерден тұрады. Олардың әрқайсысының өз ерекшеліктері болады. Атап кеткеніміздей есептеуіш кодтар калдықтар жиынтығын оқшауламайды. Бұл қағидалар жадыда көп орынды алматын өте қарапайым есептеу жұмыстарында жеңіл жүзеге асырылады. Барлығы орындалатын аймаққа және амалдардың өңделу үдерісіне байланысты орындалады.

Келесі мысалда матрицаларды бір-біріне көбейту және калдықтар жиынтығын оқшаулау программасы көрсетілген. Біз мысал ретінде матрицаларды тандадық, себебі олар көптеген ғылыми жобаларда жиі қолданылады. Матрицалар компьютерлік графика, томография генетика, криптограмма шешуі және экономика салаларында көптеген құбылыстардың тәжірибелік шешімдерін алу үшін қолданылады.

Мысал:

```
using System;
class Matrix
{
    double[,] matrix;
    int rows, columns;

    Console.WriteLine("Finalize");
}
public Matrix(int sizeA, int sizeB)
{
    rows = sizeA;
    columns = sizeB;
```

```

matrix = new double[sizeA, sizeB];
}
public double this[int i, int j]

set { matrix[i,j] = value;}
get { return matrix[i,j];}
}
public int Rows
{
get {return rows;}
}
public int Columns
{
get {return rows;}
}
}
class MatMulTest
{
static void Main(string[] args)
{
int i, size, loopCounter;
Matrix MatrixA, MatrixB, MatrixC;
size = 200;
MatrixA = new Matrix(size,size);
MatrixB = new Matrix(size,size);
MatrixC = new Matrix(size,size);
for (i=0; i<size; i++)
{
for (int j=0; j<size; j++)
{
MatrixA [i,j] = (i+j) * 10;
MatrixB [ij] = (i+j) * 20;
}
}
loopCounter = 1000;
for (i=0; i < loopCounter; i++) Matmul(MatrixA,
MatrixB, MatrixC
Console.WriteLine("Done.");
Console.ReadLine();
}
public static void Matmul(Matrix A, Matrix B, Matrix C)
{
int i, j, k, size;
double tmp;
size = A.Rows;
for (i=0; i<size; i++)
{
for (j=0; j<size; j++)
{
tmp = C[i,j];
for (k=0; k<size; k++)
{
tmp += A[i,k] * B[k,j];
}
C[ij] = tmp; }
}
}
}
}

```

Енді CLR-ден C# тіліне көшсек. Атап кеткеніміздей C# программалау тілі бірнеше программалық тілдердің жиынтығы. C# - объектіге бағытталған тіл. Біздің заманымыз өзара тығыз байланысқан объектілерден тұратын болғандықтан, объектіге бағытталған программалау тілдері ғылыми есептеу жұмыстары үшін өте ыңғайлы программалау тілі болып табылады. Сонымен қатар, құрылымдық объекті

бағытталған программалық код ғылыми модельдердің өзгерістеріне байланысты ішкі әдістер жеңіл және жылдам өзгертіле береді. Бірақ барлық ғылыми құбылыстарды объектілер және олардың өзара байланысы арқылы көрсетуге келмес - мүлдем жағдайда объектілерге нұхқзу күрделі қиындықтар туындатуы мүмкін. Мысал ретінде молекулярлық динамиканы алуға болады. Молекулярлық динамика - есептеуші химия, физика, биология және материалтану ғылымдарында кең қолданылады. Ол компьютерлерді ғылыми қолданудың алғашқы аймағы болды: 1957 жылы Элдер (Alder) және Уэйнрайт (Wainwright) аргондардың шамамен 150 атомының қозғалысын пішіндеген. Молекулярлық динамикада ғалымдарды атомдардың өзара қатынасын жұптық потенциалдар (pairwise potentials) арқылы модельдеу қызықтырады, бұған мысал ретінде күннің, планетаның жұлдыздардың өзара қатынасына гравитацияның әсерін алуға болады. Нәтижесінде жұптық күш пен энергияның есептеуге арналған теңдеуі күрделі болады, сондықтан оны есептеу үшін объектіге бағытталған жүйелерді қолданудан бас тартып дәстүрлі есептеу үрдісін пайдалануға тура келеді. Ал дәстүрлі үрдістер коды да өндірісте қолайсыз болуы мүмкін. Мұның барлығы мәліметтерді сақтау және қолдану алгоритмдерінен тәуелді болады. Осы есептеу жұмыстарын C# тілінде шығаруға болады. Әрине объектіге бағытталған программалау тілін пайдалану тиімді шешім болмауы мүмкін, бірақ ғылыми программалау бастамалары үшін қолайлы болып келеді. Нәтижелерді алу және есептеу жұмыстарын жүргізуге арналған барлық айнымалылар мен әдістерге жалғыз класс қолданылады.

C# тілінде сілтемелік (reference types) («тяжеловесные» объекті) және шамалар типі немесе мәндік типтер (value types) («облегченные» объекті) болып бөлінетін екі негізгі категория бар. Сілтемелік типтерге жады үйінділерден (жалғыз ерекшелік (кателік) — stackalloc кілттік сөзін қолдану) бөлінеді; олар абстракцияның қосымша деңгейлерін құрады, яғни олардың сақталған орнына сілтеме арқылы енді талап етеді. Бұл типтерге тікелей қатынас жасауға болмайтындықтан сілтемелік типтің айнымалылары дерлік барлық кезде сілтемені нақты объектіге (немесе null) сақтайды. Жады үйіндіден бөлінетін болғандықтан есептеу жұмыстары жүргізілетін орта әрбір ерекшелінген сұраныстың дұрыстығына сенімді болуы керек. Жадыны сәтті ерекшелетін келесі мысалды қарастырайық:

```
Matrix m = new Matrix(100, 100)
```

CLR жадысының байланыс құралын басқарушы ерекшелуге сұраныс алады, объектілерді, класс айнымалыларын сақтауға арналған жады көлемін есептейді. Содан соң жады байланыс құралын басқарушы үйіндіде есептеу жұмыстарын сақтауға арналған орын бар жоғын тексереді. Егер нысандарды сақтауға арналған орын жеткіліксіз болса, онда орынды босату және үйінділерді сығу үшін қалдықтар жиынтығы іске қосылады.

Егер есептеу үрдісі сәтті өтсе, онда жады байланыс құралын басқарушы объект жадыға сақталмас бұрын өте маңызды амалды қолдануы қажет. Ол қалдықтар жиынтығын ұрпақтан ұрпаққа (generational garbage collection) қолдану үшін қажетті және жазбалар тосқауылында (write barrier) блок коды туындайды. Өз кезегінде амалдарды орындаушы орта жазбалар тосқауылын объект жадыдағы нақты адреске жазылған сайын немесе объект жадыдағы басқа объектіге сілтеме жасалған кезде генерациялайды. Ұрпақтар бойышпа қалдықтар жиынының бұзылмауын есте сақтау қажетті бөліктердің бірі - өзге объектілерден осы объектіге сілтеме жасалған уақытта объектілерде осы жазбаның қателіксіз жинақталған болуы өте маңызды. Жазбалар тосқауылы амалдардың орындалу кезеңінде азғана кідірісте болады, сондықтан бірнеше нысандарды құру ғылымшы қосымшалар үшін тиімсіз болып табылады.

Мәндік типтер стекте сақталады. Мұндай типтерге абстракцияның қосымша деңгейлері қажет емес, сондықтан мәндік типтердің айнымалылары барлық тек сан мәнін сақтайды (сондықтан null мәніне ие болмайды). Мәндік типтердің сілтемелік типтермен салыстырғандағы басты ерекшелігі онда құрылған қосымшалар кідірісте болмайды. Оларға жады стектің қарапайым көрсеткіші арқылы стектен бөлінеді. Бұл объектілер ешқашан қалдықтар жиынтығына бастамашылық етпейді. Сонымен қатар мәндік типтер үшін жазбалар тосқауылы туындалмайды.

C# тілінде мәндік типтердің мысалы болып - мәліметтердің қарапайым типтері (int, float, single, byte), санауыштар және құрылымдар болып табылады. Сілтемелік типтің ішінде орналасқан мәлімет типі стекте сақталады. Мысалы, келесі кодқа назар аударайық:

```
class Point
{
private double x, y;
public Point(double x, double y)
{
this.x = x; this.y = y;
}
}
```

Бұл класс көшірмесі 24 байттан тұрады, оның ішінде 8 байт объектінің бас тақырыбына бөлінеді, ал қалған 16 байты - double: x және y типіндегі екі айнымалыға бөлінеді. Сонымен қатар сілтемелік типті мәндік тип объектісінің ішіне орнатқанымен барлық нысан үйіндіге орнатылмайды. Үйіндіде тек массив қана ерекшеленеді, сілтеме стекте орналасады.

Мәндік типтер өз кезегінде System.Object мұраланатын System.ValueType мұраланады. Сондықтан мәндік типтер кластарда бар функционалдылықты қамтамасыз етеді. Онда құрылымдауыштар, әдістер,

..-каторлар және артық салмактанған операторлар брлуы мүмкін, они сонымен қатар интерфейс гер реге асырылуы мүмкін. Бірақ олардан мұраланбайды, және олар өзге типтерден мураланбайды.

Ғылыми есептеу жұмыстары үшін мәндік типтер; сілтемелік типтерге Караганда жылдамырақ, тиімдірек орындалады. Көптеген өндірісшілер .NET Framework пайдаланып кітапханаларды қуруға немесе технологиялар және программалық тілдер негізінде шешуге уақыттарын және еңбек қорларын жұмсады. .NET қосымшасындағы Common Language Specification, (CLS) және Common Type System (CTS) спецификасының арқасында программалық тілдердің арақатынасын біріктіреді.

Өндірісшілер интернет компоненттерін және кең таралған қосымшаларды құру үшін C# қолданады, ал бұл программалау тілінің мүмкіндіктері өте көп, сондықтан оны ғылыми тіл ретінде пайдалануға болады.

#### Әдебиеттер

1. Петгольц Ч. Программирование в тональности C#/ Ч.Петгольц; пер. с англ. под ред. Ю.П. Леоновой. - М.: Издательско-торговый дом «Русская редакция», 2004. - 512 с: ил.
2. Троелсен Э. C# и платформа .NET. Библиотека программиста/ Э.Троелсен; пер. а англ. Р.Михеев. - СПб.Литер, 2004. - 796 с: ил.
3. Фахад Г. C# и наука: применение языковых средств C# в проектах для научных вычислений [Электронный ресурс] // MSDN Magazine. Электрон, дан. URL: <http://msdn.microsoft.com/ru-ru/library/dd353137.aspx>, свободный. Яз. рус. (дата обращения 12.12.2011).
4. Шилдт Г. Полный справочник по C#/ Г.-Шилдт : пер. с англ. - М.: Издательский дом «Вильямс», 2004. - 752 с.-ил.

УДК 504.06:614.8.06

#### МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ РАСПРЕДЕЛЕНИЯ КОНЦЕНТРАЦИИ ГАЗОВОЙ ПРИМЕСИ В АТМОСФЕРЕ ПРИ ЗАПОВОМ ВЫБРОСЕ

Наметова Ю. М., Исмаилов Б. Р., Гавриков В. В.  
ЮКГУ им. М. Ауэзова, г. Шымкент, Казахстан

#### Түйін

*Ыстық газдың бұлтының өрлеуінің үдерісі сияқты екі көріністің тізбектілігін қарастырылады Бірінші көріністе негізгі қозғаушының штиінің бұлтың айырым температур асы және қоршағт ортаиыц болып табылады. Кейін қызудың біте- шоғырланудың таратушылык концентрациясы турбулент диффузияның теңдеумен моделденеді.*

*Жұмыстың түрлі әдістерін пайдаланып, мынадай сияқты, О.М. Белоцерковский, В.А. Андрющенко, Ю.Д. Шевелев « Аламың атмосферада аяның құйынды ағысының серпіндікікі термиканың көтер-цалыбы тартудың қағидасынан» деген алған жұмыста көрсетілген. Бұдан әрі келтір- цысымның мағынасының негіздерін, нығыздықтар және сәрсенбінің цызуларынын цалыца (1) -(5) кестепік мағыналарды, ISA халықаралық стандартты атмосферасының кестелерін аппроксимация пайдаланды. Сайып келгенде ап соңғы-айырмашылық теңдеуінің жүйесін, аппроксимациялау тақырыптарды (12) • (17) прогонка әдісімен шешіледі.*

*Бұлақты создер: термик, математикалык цалып, турбулент диффузия*

#### Summary

*This paper describes the process of lifting the heated gas , as a sequence of two phenomena . In the first appearance of the main force is the temperature difference between the cloud and the environment. Once described adjustment process temperature and cloud environment, the spread of contaminants occurs solely by molecular diffusion and wind power. Various methods of work, such as , Bilotserkivskiy OM Andryushchenko VA, Sheveleva YD "Dynamics of spatial vortex flows in an inhomogeneous atmosphere ," according to which model the therm uplift derived from the theory of dimensions is based on several assumptions, which are described in the work. 7> following is a substitution of values of pressure, density and temperature of the medium in the model (1) • (5), approximate the table values using Table International Standard Atmosphere ISA. As a result, we obtain a system finite difference equations that approximate the problem (12) - (17) is solved by the sweep method .The process lifting the clouds of hot gas is seen as a sequence of two phenomena. In the first main driving force is tk temperature difference between the cloud and the environment. After leveling the temperature distribution oft concentration is modeled by the equation of turbulent diffusion.*

В работе создана упрощенная математическая модель подъема облака нагретого газа (термика) по кратковременного выброса примеси большой мощности, с целью оценки динамических параме подъема термика до выравнивания его температуры и скорости относительно окружающей среды [5, с. 26]. Предполагается, что после процесса уравнивания температуры облака и окружающей с