

**ЕФЕКТИВНІСТЬ АНАЛІЗУ СТУДЕНТАМИ МОВ ПРОГРАМУВАННЯ C++ ТА C#
У ПРОЦЕСІ ВИКОНАННЯ КУРСОВИХ РОБІТ З МЕТОЮ ОЦІНКИ ПЕРЕВАГ
ТА НЕДОЛІКІВ РІЗНИХ МОВ ПРОГРАМУВАННЯ ТА ОБҐРУНТУВАННЯ
ВИБОРУ НЕОБХІДНОЇ МОВИ**

Розглядається задача оцінювання переваг та недоліків різних мов програмування та обґрунтування вибору необхідної мови для виконання конкретної поставленої математичної задачі. Для цього виконується курсова робота з навчальної дисципліни "Алгоритми та структури даних", в ході якої студенти мають порівнювати способи та принципи роботи з мовою програмування C++ на відміну від цих же засобів в мові C#. В результаті наводиться розмежування областей ефективного використання мов C++ та C#.

Ключові слова: об'єктно-орієнтоване програмування, розумові структури, самоосвіта, пізнавальна діяльність, чисельні методи, кореляційний та регресивний аналіз.

N. I. PRAVORSKA
Khmelnitskiy National University

**EFFICIENCY OF STUDENTS' ANALYSIS OF PROGRAMMING LANGUAGES C++ AND C#
ALONG WITH THE COURSE WORK IMPLEMENTATION TO ASSESS ADVANTAGES AND DISADVANTAGES
OF DIFFERENT PROGRAMMING LANGUAGES AND FOR A RATIONALE
IN SELECTING AN APPROPRIATE LANGUAGE**

The task of evaluating advantages and disadvantages of different programming languages and making a rationale for selecting an appropriate language for a particular mathematical problem is considered. For this, the course work by the subject "Algorithms and Data Structures" is executed, in which students have to compare the methods and principles of the programming language C++ as opposed to the same tools in C#. As a result, a delimitation of efficient-use fields of languages C++ and C# is given.

Keywords: object-oriented programming, mental structure, self-education, cognitive activity, numerical methods, correlation and regression analysis.

Актуальність та постановка задачі. Об'єктно-орієнтоване програмування (ООП) на сьогоднішній день займає важливе місце в курсах вивчення інформатики та програмування вищих навчальних закладів. Вивчення такого матеріалу сприяє повній підготовці висококваліфікованих спеціалістів, які здатні швидко адаптуватися до сучасних об'єктно-орієнтованих технологій і є базовим інструментом розвитку інтелектуальних здібностей людини, яка вважається складовою її культурологічного розвитку.

Програмування – це не тільки власне написання програм. Ефективність діяльності програміста сильно залежить від оточуючого середовища, в якій вона здійснюється. Це середовище включає в себе набір інструментальних засобів, які складають *систему програмування* і тісно взаємодіють з операційною системою. Це середовище включає в себе також і виробничі умови, моральний мікроклімат і т.п. В подальшому, на основі синтаксико-семантичної моделі психологи вважають, що в процесі вивчення програмування у мозку формуються багаторівневі *семантичні розумові структури*, які не пов'язані з визначеною мовою [1, с.24]. Саме тому зазвичай складніше за все вивчити першу мову програмування: доводиться поєднувати вивчення синтаксису з формуванням розумових структур. Без цього неможливо добитися розуміння, хоча, як і в математиці, необхідний відомий компроміс між розумінням та результатом.

Метою цієї статті є забезпечення ефективності аналізу студентами мов програмування C++ та C# у процесі виконання курсових (творчих) робіт з використанням цих мов для того, щоб вони могли оцінити переваги та недоліки різних мов програмування та навчитися обирати необхідну мову для виконання конкретної поставленої математичної задачі.

З точки зору психології, складання програм є багатоетапним процесом, під час якого від формулювання задачі шляхом покрокового уточнення або переформулювання програміст приходять до розв'язку задачі у вигляді внутрішніх семантичних структур, а потім вже і до розв'язку задачі на заданій мові програмування. Процес розуміння програми – її переведення у внутрішню семантичну структуру. Навчання програмуванню – безперервний процес, внутрішньо властивий самій професії програміста.

Також важливим є процес самоосвіти, можливість знайти необхідні для розв'язання поставленої задачі дані в постійно зростаючому обсязі інформації. Сьогодні кожний студент, який отримує вищу освіту, повинен мати конкретні навички з використанням технічних засобів (в тому числі комп'ютерних мереж); вміти 1) добувати інформацію з різних джерел – як із періодичної літератури, так і з електронних комунікацій; 2) переробляти отриману інформацію; 3) представляти її в зрозумілому вигляді; 4) ефективно її використовувати; знати особливості інформаційних потоків у сфері своєї діяльності.

В психолого-педагогічній літературі [2, с.515] з проблеми пізнавальної самостійності наголошується на тому, що пізнавальна самостійність – це одна з властивостей особистості, яка

характеризується такими основними факторами: по-перше, сукупністю знань, умінь та навичок, якими володіє особистість; по-друге, відношенням особистості до процесу діяльності, до результатів і умов її виконання, а також зв'язками з іншими людьми, що виникають в процесі цієї діяльності.

Пізнавальна самостійність проявляється в самостійній пізнавальній діяльності, і від викладача залежить правильна організація такої діяльності. Існують різні підходи до практичного формування пізнавальної самостійності в навчанні [3, с.83].

Під час вивчення студентами дисципліни “Алгоритми та структури даних” враховувалося, що студенти вже проходили курс програмування з вивченням мови ООП С#, а також ними була виконана курсова робота з побудовою математичної моделі запропонованої задачі і написанням програми на мові С#, а також створення проекту з підключенням розроблених баз даних в Microsoft Access. Тому в ході вивчення мови програмування С++, яка була покладена в основу вивчення курсу “Алгоритми та структури даних”, студенти мали змогу порівнювати програмні та алгоритмічні конструкції цих провідних мов програмування, оцінити узагальнене програмування за рахунок використання шаблонів цих мов.

Під час виконання наступної курсової роботи студентам було запропоновано створити математичну модель для розв'язання запропонованої математичної задачі, а саме: основних чисельних методів; спектрального, статистичного, кореляційного та регресивного аналізу; обчислення спеціальних функцій (Бесселя, гамма-функцій, Вебері і т. ін). Написати програму на мові С++, але використовувати не готові програмні об'єкти (кнопки, таблиці, вікна редагування і т. ін.) Visual Studio, як це було в курсовій роботі з використанням мови С#, а написати та відформатувати їх самим.

Важливість використання прикладних задач визначається роллю цих задач в розвитку пізнавального інтересу студентів, їх творчих можливостей, самостійності, гнучкості розуму, умінь узагальнювати знання з різних предметів і наук, а також в розвитку інформаційної культури студентів.

Прикладна задача – теоретична задача практичного змісту [4, с.72]. Практична задача – це задача, в якій описується практико-орієнтована ситуація і розв'язання якої вимагає визначених практичних навичок, в тому числі використання засобів інформаційних технологій.

Тому в ході опанування мови програмування С++ та дисципліни “Алгоритми та структури даних” поєднання наступних форм самостійних робіт якнайбільше сприяє розвитку пізнавальної самостійності студентів:

1. Самостійні роботи, які виконуються за інструкцією (це звичайні задачі на лабораторних роботах). Вони носять пізнавально-узагальнюючий характер, тому що відбувається формування нових понять та навичок роботи з мовою програмування С++. Роль викладача – консультація студентів протягом виконання роботи, подолання ускладнень, які виникають у студентів.

2. Самостійні роботи індивідуального характеру (запропонована розрахункова робота). Вони носять систематизуючий характер, тому що відбувається засвоєння основних положень математичних розрахунків, основних технологій роботи зі знаходження необхідної інформації в мережі Internet та написання програми мовою програмування С++. В основі цих робіт – продуктивний спосіб діяльності. Роль викладача – згідно з індивідуальними або професійними інтересами студентів підготувати завдання [5, с.19].

Приклад поставленої задачі курсової роботи. Тема: Розробка програмної оболонки для апроксимації полінома за методом найменших квадратів.

Математична модель. Апроксимація.

Розглянемо векторний простір V , елементами якого є функції f якого-небудь класу, визначені на одній і тій же області M , і в якій визначена норма $p(f)$, яка має ті ж властивості, що й норма в евклідовому і гільбертовому просторах. Нехай f – елемент з V , і G – простір V . Тоді загальна проблема апроксимації $AP(f, G, p(f))$ полягає в наступному: знайти функцію $\hat{g}(x) \in G$ таку, що

$$D(f, G, p(f)) = \inf_{g \in G} \{p(f - g)\} = p(f - \hat{g}).$$

Функція $\hat{g}(x)$ називається найкращим наближенням, що відповідає $AP(f, G, p(f))$, число $D(f, G, p)$ – дефектом.

З практичної точки зору, мова йде про заміну функції f , що чисельно не виражається через функцію $\hat{g}(x)$, що виражається, за можливістю, як найточніше, тобто, так, щоб для заданого дійсного числа $\varepsilon > 0$ підходящим вибором G одержати дефект, менший за ε .

При обчисленні $AP(f, G, p(f))$ повинні бути знайдені відповіді на такі запитання:

- 1) чи існує найкраще наближення \hat{g} ?
- 2) чи однозначно визначено \hat{g} ?
- 3) як можна обчислити \hat{g} ?

Метод найменших квадратів.

Розглянемо V – простір інтегрованих функцій, визначених на $[a; b]$, з інтегрованим квадратом.

Скалярний добуток функцій $f(x)$ та $g(x)$ визначається як $\int_a^b f(x)g(x)dx$. Такий простір іноді позначається

$R_2(a, b)$. Нехай G_{n+1} – множина всіх многочленів P_n з дійсними коефіцієнтами степені не більше n .

Сформулюємо задачу $AP(f, G_{n+1}, \|\cdot\|)$ наступним чином.

Для заданої функції f знайти многочлен

$$P_n = \hat{a}_0 + \hat{a}_1 x + \hat{a}_2 x^2 + \dots + \hat{a}_n x^n$$

такий, щоб для всіх многочленів P_n із G_{n+1} виконувалась нерівність:

$$\int_a^b (f(x) - \hat{P}_n(x))^2 dx \leq \int_a^b (f(x) - P_n(x))^2 dx .$$

Набір $h_k = x^{k-1}$ ($k = 1, 2, \dots, n+1$) степенів функцій виступає як єдиний базис G_{n+1} , а коефіцієнти $\hat{a}_0, \hat{a}_1, \dots, \hat{a}_n$ мінімального многочлена виражаються із системи рівнянь:

$$\sum_{l=0}^n \hat{a}_l \int_a^b x^{l+k-2} dx = \int_a^b x^{l-1} f(x) dx, \quad (k = 1, 2, \dots, n+1);$$

$$\sum_{l=0}^n \hat{a}_l \frac{1}{l+k-1} (b^{l+k-1} - a^{l+k-1}) = \int_a^b x^{k-1} f(x) dx, \quad (k = 1, 2, \dots, n+1).$$

Слід відмітити, що вона не дуже зручна для обчислень, і може обчислювати з невеликою точністю. Наприклад, при $a=0$ та $b=1$ матрицею цієї системи є матриця Гілберта:

$$H_n = \begin{bmatrix} 1/2 & 1/2 & \dots & 1/n \\ 1/2 & 1/3 & \dots & 1/(n+1) \\ \dots & \dots & \dots & \dots \\ 1/n & 1/(n+1) & \dots & 1/(2n-1) \end{bmatrix},$$

норма якої зі збільшенням n необмежено зростає.

Система рівнянь стає особливо простою, якщо в якості базису $\{h_1, h_2, \dots, h_n\}$ просторів G_{n+1} обрати ортогональні многочлени. Для таких многочленів $Q_i(x)$ справедлива рівність:

$$\int_a^b Q_i(x) Q_j(x) dx = 0 \quad (i \neq j)$$

Якщо, крім того,

$$\int_a^b Q_i^2(x) dx = 1 \quad (i = 0, 1, \dots, n),$$

то говорять, що многочлени утворюють ортогональну систему.

Для відрізка $[a; b] = [-1; 1]$ – це відомий многочлен Лагранжа

$$P_n(x) = \frac{1}{2^n (n!)} \cdot \frac{d^n}{dx^n} (x^2 - 1)^n, \quad P_0(x) \equiv 1,$$

з властивістю ортогональності

$$\int_{-1}^1 P_n(x) P_m(x) dx = 0, \quad (m \neq n),$$

$$\int_{-1}^1 (P_n(x))^2 dx = \frac{2}{2n+1},$$

де відома рекурентна формула:

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)$$

Многочлен $P_n(x)$ має на сегменті $[-1; 1]$ рівно n різних дійсних коренів. Подальше співвідношення ортогональності є таким:

$$\int_{-1}^1 x^k P_n(x) dx = 0, \quad (k = 0, 1, \dots, n-1),$$

$$\int_{-1}^1 x^k P_n(x) dx = \frac{2^{n+1} (n!)^2}{(2n+1)!}.$$

Якщо найкраще наближення $\hat{P}_n(x)$ на сегменті $[-1; 1]$ шукається у вигляді лінійної комбінації многочленів Лагранжа

$$\hat{P}_n(x) = \sum_{k=0}^n c_k^{(n)} P_k(x),$$

то для відшукування коефіцієнтів $c_k^{(n)}$ система рівнянь суттєво спрощується:

$$c_k^{(n)} \equiv c_k = \frac{2k+1}{2} \int_{-1}^1 f(x) P_k(x) dx,$$

$$D^2(f, G_{n+1}, \| \cdot \|) = \|f\|^2 - \sum_{k=0}^n \frac{2k+1}{2} (f, P_k)^2 = \int_{-1}^1 f^2(x) dx - \left(\sum_{k=0}^n \left[\int_{-1}^1 f(x) P_k(x) dx \right]^2 \frac{2k+1}{2} \right).$$

Приклад апроксимації многочлену методом найменших квадратів.

Знайти параболу, яка найкращим чином апроксимує функцію $y = \sin t$ на відрізку $[0, \pi]$ в значенні квадратичного відхилення.

1. Перетворення відрізка:

$$x = \frac{2t}{\pi} - 1, \quad y = \sin \left[\frac{\pi}{2}(x+1) \right], \quad -1 \leq x \leq 1.$$

2. Многочлени Лагранжа:

$$P_0 = 1, \quad P_1(x) = x, \quad P_2(x) = \frac{1}{2}(3x^3 - 1).$$

3. Коефіцієнти розкладу:

$$c_0 = \frac{1}{2} \int_{-1}^1 \sin \left(\frac{\pi}{2}(x+1) \right) dx = \frac{2}{\pi},$$

$$c_1 = \frac{3}{2} \int_{-1}^1 x \sin \left(\frac{\pi}{2}(x+1) \right) dx = 0,$$

$$c_2 = \frac{5}{2} \int_{-1}^1 (3x^3 - 1) \cdot \frac{1}{2} \sin \left(\frac{\pi}{2}(x+1) \right) dx = \frac{10}{\pi} \left(1 - \frac{\pi}{12} \right).$$

4. Апроксимація [6, с. 310]:

$$\hat{P}_2(x) = \frac{\pi}{2} + \frac{10}{\pi} \left(1 - \frac{12}{\pi^2} \right) \cdot \frac{1}{2} (3x^3 - 1)$$

або

$$\hat{P}_2(x) = \frac{2}{\pi} + \frac{10}{\pi} \left(1 - \frac{12}{\pi^2} \right) \cdot \left(\frac{6}{\pi^2} \left(t - \frac{\pi}{2} \right)^2 - \frac{1}{2} \right).$$

Виконання програми.

Програма являє собою оболонку для апроксимації многочлена методом найменших квадратів. Для обчислення необхідно ввести кількість вузлів апроксимації (рис. 1). Залежно від цієї кількості, необхідно буде ввести стільки ж пар чисел X та Y (рис. 2).

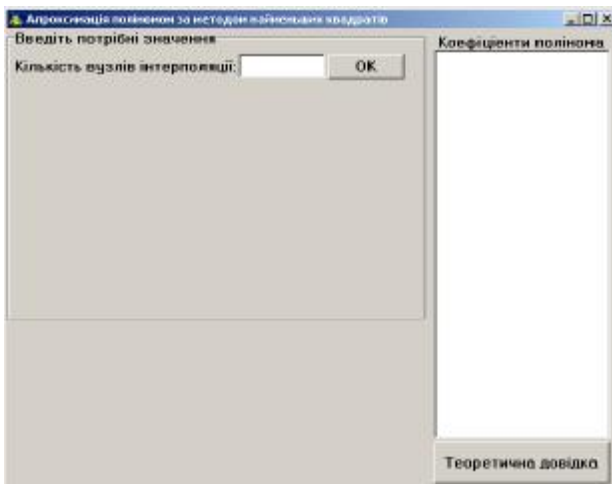


Рис. 1. Основна форма програми

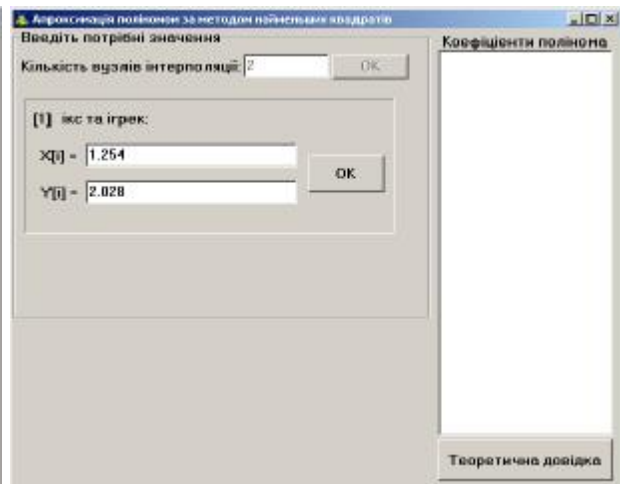


Рис. 2. Введення чисел X та Y

Необхідно ввести або значення похибки інтерполяції, або степінь полінома. Якщо буде введено значення похибки інтерполяції, степінь полінома буде обчислюватися за формулою. І навпаки: якщо буде введено степінь полінома, – розрахується значення похибки інтерполяції. Для вибору необхідного шляху обчислення, необхідно мишкою поставити одну з двох відміток (рис. 3).

Після ведення цих даних, буде проведена частина обрахунків. Якщо дані були введені правильно – необхідно ввести ще значення X (рис. 4).

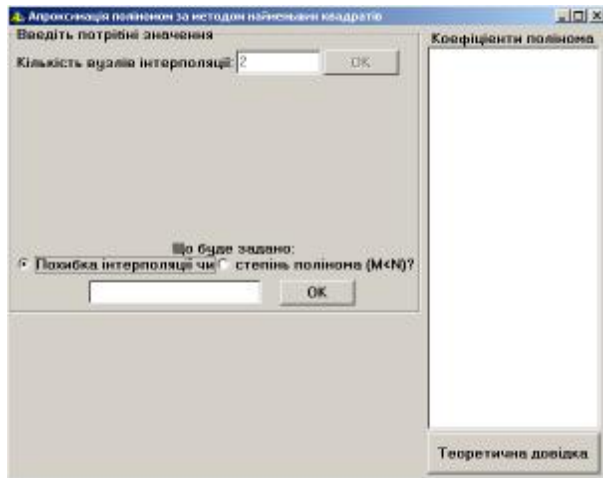


Рис. 3. Вікно вибору варіантів розрахунку

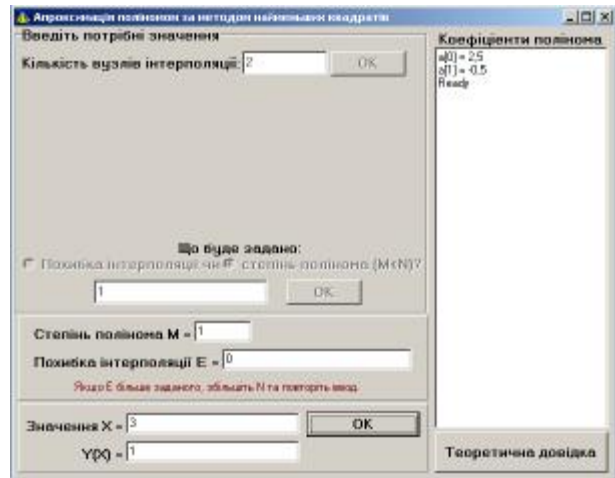


Рис. 4. Вікно для введення значення X

Після цього буде обчислено значення Y.

У вікні “коефіцієнти полінома” з’являться числа a[0], a[1]. Це і є коефіцієнти полінома.

Натиснувши кнопку “Теоретична довідка”, відкриється нове вікно (info.txt). У цьому файлі містяться короткі теоретичні відомості про апроксимацію полінома методом найменших квадратів та про метод найменших квадратів (рис. 5).

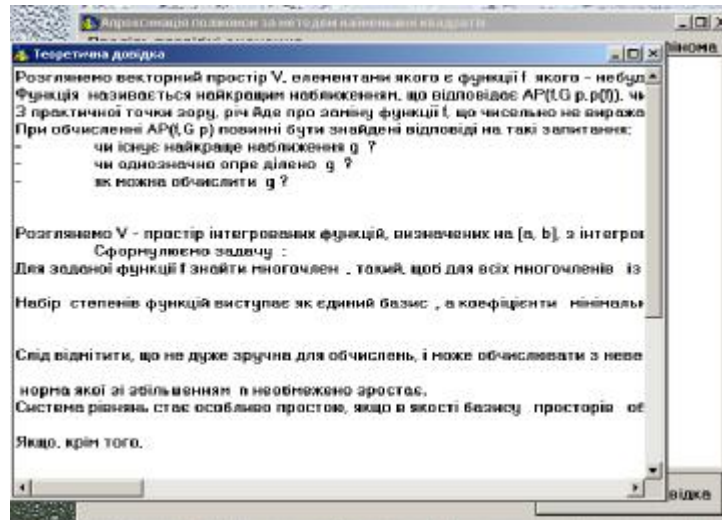


Рис. 5. Вікно довідки

Усі віконця програми зроблені на одній формі. Вони знаходяться в стані invisible до тих пір, поки не прийде черга для їх активізації. При відкритті “нового” вікна (visible), “старе” вікно стає invisible. Це забезпечує простоту програми і не вимагає переписання щоразу нової форми, що дещо прискорює обрахунки.

Представлений лістинг програми – файл Unit1.cpp та налаштування самої форми – файл Unit1.dfm. Налаштування об’єктів, які є в наданому файлі Unit1.dfm, не використовувались при написанні програми мовою C# і розробляються студентами самостійно.

<pre>Unit1.cpp //----- #include <vcl.h> #pragma hdrstop #include "Unit1.h" #include "Unit2.h" //----- #pragma package(smart_init)</pre>	<pre>g[k]=b[k]/a[k][k]; k1=k+1; for (i=k1;i<=n;i++) { b[i]=a[i][k]*g[k]; for (j1=k;j1<=n;j1++) { j=n-j1+k; c[k1][j]=a[k][j]/a[k][k]; a[i][j]=a[i][k]*c[k1][j];</pre>
---	--

<pre>#pragma resource "*.dfm" TForm1 *Form1; //----- __fastcall TForm1::TForm1(TComponent* Owner) : TForm(Owner) { } //----- void __fastcall TForm1::Button4Click(TObject *Sender) { Form2->Visible=true; } //----- void __fastcall TForm1::Button2Click(TObject *Sender) { h=StrToInt(Edit2->Text); i=1; Button2->Enabled=false; Edit2->Enabled=false; Panel2->Visible=true; } //----- void __fastcall TForm1::Button1Click(TObject *Sender) { if (RadioButton2->Checked==true) n=StrToInt(Edit1->Text); else { e1=StrToFloat(Edit1->Text); n=1; }; Button1->Enabled=false; RadioButton1->Enabled=false; RadioButton2->Enabled=false; Edit1->Enabled=false; again: n++; for (j=0;j<=2*n-1;j++) if (j>n) d[j]=0; else {b[j]=0; d[j]=0;}; for (i=1;i<=h;i++) { r=y[i]; f=1; for (j=1;j<=2*n-1;j++) if (j>n) {d[j]+=f; f*=x[i];} else {b[j]+=r; r*=x[i]; d[j]+=f; f*=x[i];}; }; for (i=1;i<=n;i++) { k=i; for (j=1;j<=n;j++) { a[i][j]=d[k]; k++; }; }; n1=n-1; for (k=1;k<=n1;k++) { if (fabs(a[k][k])<=0) { k1=k+1; for (m=k1;m<=n;m++) if (fabs(a[m][k]!=0) for (l=1;l<=n;l++) { u=a[k][l]; a[k][l]=a[m][l]; a[m][l]=u; }; u=b[k]; b[k]=b[m]; b[m]=u; }; }; };</pre>	<pre>}; }; m=n; z[m]=b[m]/a[m][m]; do { m--; s=0; for (l=m;l<=n;l++) s+=c[m][l+1]*z[l+1]; z[m]=g[m]-s; } while (m>1); e=0; for (i=1;i<=h;i++) { s=y[i]; r=1; for (j=1;j<=n;j++) { s=r*z[j]; r*=x[i]; }; e+=s*s; }; e=sqrt(e/h); if ((RadioButton2->Checked==true) (e<=e1) { Panel1->Visible=true; Edit5->Text=IntToStr(n-1); Edit6->Text=FloatToStr(e); for (i=1;i<=n;i++) { Memo1->Lines->Add("a["+IntToStr(i-1)+"] = "+FloatToStr(z[i])); }; Panel3->Visible=true; } else if (n<h) goto again; Memo1->Lines->Add("Ready"); } //----- void __fastcall TForm1::Button3Click(TObject *Sender) { if (i<=h) { x[i]=StrToFloat(Edit3->Text); y[i]=StrToFloat(Edit4->Text); i++; Label3->Caption="["+IntToStr(i)+"]"; Edit3->Text=""; Edit4->Text=""; }; if (i>h) { Panel2->Visible=false; RadioButton1->Visible=true; RadioButton2->Visible=true; Label1->Visible=true; Edit1->Visible=true; Button1->Visible=true; }; } //----- void __fastcall TForm1::Button5Click(TObject *Sender) { X=StrToFloat(Edit7->Text); q=0; for (i=n;i>=2;i--) q=(q+z[i])*X; q+=z[1]; Edit8->Text=FloatToStr(q); } //-----</pre>
--	--

<pre>Unit1.dfm object Form1: TForm1 Left = 192 Top = 114 BorderStyle = bsSingle Caption = #1040#1087#1088#1086#1082#1089#1080#1084#1072#1094#1110#1 103' #1087#1086#1083#1110#1085#1086#1084#1086#1084' #1079#1072' #1084#1077#1090#1086#1076#1086#1084' #1085#1072#1081#1084#1077#1085#1100#1096#1080#1093' #1082#1074#1072#1076#1088#1072#1090#1110#1074 ClientHeight = 450 ClientWidth = 600 Color = clBtnFace Font.Charset = DEFAULT_CHARSET</pre>	<pre>Height = 25 Caption = #1054#1050 TabOrder = 3 Visible = False OnClick = Button1Click end object Edit2: TEdit Left = 228 Top = 24 Width = 85 Height = 24 TabOrder = 4 end object Button2: TButton Left = 316</pre>
---	--

```

Font.Color = clWindowText
Font.Height = -11
Font.Name = 'MS Sans Serif'
Font.Style = []
OldCreateOrder = False
PixelsPerInch = 96
TextHeight = 13
object Label7: TLabel
  Left = 424
  Top = 4
  Width = 164
  Height = 16
  Caption =
'#1050#1086#1077#1092#1110#1094#1110#1077#1085#1090#1080'
'#1087#1086#1083#1110#1085#1086#1084#1072'
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -13
  Font.Name = 'MS Sans Serif'
  Font.Style = [fsBold]
  ParentFont = False
end
object GroupBox1: TGroupBox
  Left = 0
  Top = 0
  Width = 413
  Height = 285
  Caption = '#1042#1074#1077#1076#1110#1090#1100'
'#1087#1086#1090#1088#1110#1073#1085#1110'
'#1079#1085#1072#1095#1077#1085#1085#1103'
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -13
  Font.Name = 'MS Sans Serif'
  Font.Style = [fsBold]
  ParentFont = False
  TabOrder = 0
  object Label1: TLabel
    Left = 164
    Top = 212
    Width = 124
    Height = 16
    Caption = '#1065#1086' '#1073#1091#1076#1077'
'#1079#1072#1076#1072#1085#1086:'
    Font.Charset = DEFAULT_CHARSET
    Font.Color = clWindowText
    Font.Height = -13
    Font.Name = 'MS Sans Serif'
    Font.Style = [fsBold]
    ParentFont = False
    Visible = False
  end
  object Label2: TLabel
    Left = 8
    Top = 28
    Width = 219
    Height = 16
    Caption = '#1050#1110#1083#1100#1082#1110#1089#1090#1100'
'#1074#1091#1079#1083#1110#1074'
'#1110#1085#1090#1077#1088#1087#1086#1083#1103#1094#1110#111:'
  end
  object Panel2: TPanel
    Left = 12
    Top = 68
    Width = 365
    Height = 133
    BevelOuter = bvLowered
    TabOrder = 6
    Visible = False
    object Label6: TLabel
      Left = 16
      Top = 85
      Width = 41
      Height = 16
      Caption = 'Y[i] = '
    end
    object Label5: TLabel
      Left = 16
      Top = 48
      Width = 40
      Height = 16
      Caption = 'X[i] = '
    end
    Top = 24
    Width = 75
    Height = 25
    Caption = '#1054#1050'
    TabOrder = 5
  end
  object Panel1: TPanel
    Left = 4
    Top = 288
    Width = 409
    Height = 85
    TabOrder = 1
    Visible = False
    object Label8: TLabel
      Left = 16
      Top = 10
      Width = 158
      Height = 16
      Caption = '#1057#1090#1077#1087#1110#1085#1100'
'#1087#1086#1083#1110#1085#1086#1084#1072' '#1052' = '
      Font.Charset = DEFAULT_CHARSET
      Font.Color = clWindowText
      Font.Height = -13
      Font.Name = 'MS Sans Serif'
      Font.Style = [fsBold]
      ParentFont = False
    end
    object Label9: TLabel
      Left = 16
      Top = 38
      Width = 190
      Height = 16
      Caption = '#1055#1086#1093#1080#1073#1082#1072'
'#1110#1085#1090#1077#1088#1087#1086#1083#1103#1094#1110#1111' '#1045' = '
      Font.Charset = DEFAULT_CHARSET
      Font.Color = clWindowText
      Font.Height = -13
      Font.Name = 'MS Sans Serif'
      Font.Style = [fsBold]
      ParentFont = False
    end
    object Label10: TLabel
      Left = 56
      Top = 64
      Width = 285
      Height = 13
      Caption = '#1071#1082#1097#1086' '#1045'
'#1073#1110#1083#1100#1096#1077'
'#1079#1072#1076#1072#1085#1086#1075#1086',
'#1079#1073#1110#1083#1100#1096#1110#1090#1100' 'N
'#1090#1072'
'#1087#1086#1074#1090#1086#1088#1110#1090#1100'
'#1074#1074#1086#1076:'
      Font.Charset = DEFAULT_CHARSET
      Font.Color = clMaroon
      Font.Height = -11
      Font.Name = 'MS Sans Serif'
      Font.Style = []
      ParentFont = False
    end
  end
  object Edit5: TEdit
    Left = 176
    Top = 5
    Width = 57
    Height = 21
    ReadOnly = True
    TabOrder = 0
  end
  object Edit6: TEdit
    Left = 204
    Top = 33
    Width = 189
    Height = 21
    ReadOnly = True
    TabOrder = 1
  end
  object Button4: TButton
    Left = 420
    Top = 404
  end

```

<pre> end object Label4: TLabel Left = 36 Top = 12 Width = 87 Height = 16 Caption = '#1110#1082#1089' '#1090#1072' '#1110#1075#1088#1077#1082:' end object Label3: TLabel Left = 8 Top = 12 Width = 19 Height = 16 Caption = '[1]' end object Edit3: TEdit Left = 60 Top = 44 Width = 209 Height = 24 TabOrder = 0 end object Edit4: TEdit Left = 60 Top = 81 Width = 209 Height = 24 TabOrder = 1 end object Button3: TButton Left = 280 Top = 56 Width = 75 Height = 37 Caption = '#1054#1050' TabOrder = 2 OnClick = Button3Click end end object RadioButton1: TRadioButton Left = 8 Top = 228 Width = 201 Height = 17 Cursor = crHandPoint Caption = '#1055#1086#1093#1080#1073#1082#1072' '#1110#1085#1090#1077#1088#1087#1086#1083#1103#1094#1110#1111' '#1095#1080' Checked = True Font.Charset = DEFAULT_CHARSET Font.Color = clWindowText Font.Height = -13 Font.Name = 'MS Sans Serif' Font.Style = [fsBold] ParentFont = False TabOrder = 0 TabStop = True Visible = False end object RadioButton2: TRadioButton Left = 208 Top = 228 Width = 201 Height = 17 Cursor = crHandPoint Caption = '#1089#1090#1077#1087#1110#1085#1100' '#1087#1086#1083#1110#1085#1086#1084#1072' (M<N)?' Font.Charset = DEFAULT_CHARSET Font.Color = clWindowText Font.Height = -13 Font.Name = 'MS Sans Serif' Font.Style = [fsBold] ParentFont = False TabOrder = 1 Visible = False end object Edit1: TEdit Left = 80 Top = 252 Width = 173 Height = 24 TabOrder = 2 </pre>	<pre> Width = 173 Height = 41 Caption = '#1058#1077#1086#1088#1077#1090#1080#1095#1085#1072' '#1076#1086#1074#1110#1076#1082#1072' Font.Charset = DEFAULT_CHARSET Font.Color = clWindowText Font.Height = -13 Font.Name = 'MS Sans Serif' Font.Style = [fsBold] ParentFont = False TabOrder = 2 OnClick = Button4Click end object Panel3: TPanel Left = 4 Top = 376 Width = 409 Height = 69 TabOrder = 3 Visible = False object Label11: TLabel Left = 8 Top = 14 Width = 98 Height = 16 Caption = '#1047#1085#1072#1095#1077#1085#1085#1103' '#1061' = Font.Charset = DEFAULT_CHARSET Font.Color = clWindowText Font.Height = -13 Font.Name = 'MS Sans Serif' Font.Style = [fsBold] ParentFont = False end object Label12: TLabel Left = 64 Top = 42 Width = 46 Height = 16 Caption = 'Y(X) = ' Font.Charset = DEFAULT_CHARSET Font.Color = clWindowText Font.Height = -13 Font.Name = 'MS Sans Serif' Font.Style = [fsBold] ParentFont = False end object Edit7: TEdit Left = 108 Top = 9 Width = 173 Height = 21 TabOrder = 0 end object Edit8: TEdit Left = 108 Top = 37 Width = 173 Height = 21 ReadOnly = True TabOrder = 1 end object Button5: TButton Left = 288 Top = 8 Width = 113 Height = 25 Caption = '#1054#1050' Font.Charset = DEFAULT_CHARSET Font.Color = clWindowText Font.Height = -13 Font.Name = 'MS Sans Serif' Font.Style = [fsBold] ParentFont = False TabOrder = 2 OnClick = Button5Click end end object Memo1: TMemo Left = 420 Top = 20 Width = 173 </pre>
--	--

<pre>Visible = False end object Button1: TButton Left = 272 Top = 252 Width = 75</pre>	<pre>Height = 381 ReadOnly = True TabOrder = 4 end end</pre>
--	--

Отже, в процесі написання програми студенти опановують мову програмування C++, порівнюючи її з мовою C#. А також самостійно розробляють стандартні об'єкти форми.

Висновки. Після підготовки курсової роботи студенти можуть порівняти обидві мови програмування C++ та C# та самі дійти до деяких суджень, а саме: 1) в C# немає асемблера, такого ж як в C++, немає високорівневого об'єктно-орієнтованого MSIL. Отже в C# неможливо написати код, який обчислює хеш CRC32 в 4-5 рядків асемблерних команд. Його можна написати на C#, отримавши близько 45 рядків на MSIL. І поскаржитися, що при цьому виконання гальмує. І вірно. Адже навіть на C++ для цього застосовують асемблерну вставку; 2) в C# все сильно стандартизовано. Неможливо створити посилання на byte або boolean. І неможливо створити посилання на знищений об'єкт. Поки об'єкт живий, посилання буде існувати. Коли посилання не буде, то і об'єкт помре; 3) в C# виділення пам'яті відбувається за допомогою віртуальної машини. А в C++ виконується все вручну. Звичайно, якщо бінарні поля упаковані і зроблені з них об'єкти, то все буде куди компактніше ніж аналогічне в C#.

Отже, проведений ефективний аналіз дає студенту змогу зрозуміти, що C# хороший для написання дрібних програм, велика частина яких займає інтерфейс користувача, різні утиліти, навчальні завдання, АРМи, щоб дістати 3 параметра з БД. Поганий для ресурсоємних додатків, для різних промислових застосувань, де обсяг продажів малий в часі, і, на жаль, кожні 3 роки доводиться переходити на новий Windows, також існує нечітка реалізація алгоритмів в .NET. Для C++ існують величезні поклади відкритих початкових кодів по всьому Інтернету, витончені конструкції, максимально наближені до машинного коду, на будь-яку нову платформу першим ділом встановлюється C/C++ компілятор. Мінуси – при всій своїй універсальності написання користувальницьких інтерфейсів досить трудомістке і болісне. Але, зрештою, все мають обрати самі студенти після своєї порівняльної, пізнавальної, самостійної діяльності.

Література

1. Вирт Н. Алгоритмы + Структуры данных = Программы / Вирт Н. ; пер. с англ. – М. : Мир, 1985. – 406 с.
2. Педагогика. Большая современная энциклопедия / сост. Е. С. Рапацевич. – Минск : Соврем. слово, 2005. – 720 с.
3. Калашникова Л. М. Формирование познавательной самостоятельности студентов в процессе изучения специальных дисциплин / Л. М. Калашникова // Педагогическое образование и наука. – 2011. – № 7. – С. 82–85.
4. Гасс С. Путешествие в страну линейного программирования / Гасс С. ; пер. с англ. ; под ред. Ю.Н. Сударева. – М., 1973. – 175 с.
5. Бойко Н. І. Організація самостійної роботи студентів вищих навчальних закладів в умовах застосування інформаційно-комунікаційних технологій : автореф. дис. на здобуття наук. ступеня канд. пед. наук : спец. 13.00.04 "Теорія і методика професійної освіти" / Н. І. Бойко. – К., 2008. – 23 с.
6. Zhiyou Wu. Global optimality conditions and optimization methods for constrained polynomial programming problems / Zhiyou Wu, Jing Tian, Julien Ugon, Liang Zhang // Applied Mathematics and Computation. – 2015. – Volume 262. – P. 312–325.

Рецензія/Peer review : 8.1.2016 р. Надрукована/Printed :25.2.2016 р.
Стаття рецензована редакційною колегією