

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

МЕТОДИЧНІ ВКАЗІВКИ

**до виконання лабораторної роботи «Розробка проекту Delphi»
з навчальної дисципліни «Інформаційні технології»**

Харків

2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

МЕТОДИЧНІ ВКАЗІВКИ

до виконання лабораторної роботи «Розробка проекту Delphi»
з навчальної дисципліни «Інформаційні технології»
для студентів усіх форм навчання
за спеціальністю 131 «Прикладна механіка»

Затверджено
редакційно-видавничою
радою університету,
протокол №1 від 13.02.2025 р.

Харків
НТУ «ХП»

2025

Методичні вказівки до виконання лабораторної роботи «Розробка проекту Delphi» з навчальної дисципліни «Інформаційні технології» для студентів усіх форм навчання за спеціальністю 131 «Прикладна механіка» / уклад.: Т. Є. Третяк, О. Л. Мироненко, С. О. Мироненко. – Харків: НТУ «ХП», 2025. – 26 с.

Укладачі: Т. Є. Третяк
О. Л. Мироненко
С. О. Мироненко

Рецензент О. В. Кобець

Кафедра інтегрованих технологій машинобудування» ім. М. Ф. Семка

ЗМІСТ

Вступ	4
Мета виконання роботи	5
1. Формулювання завдання	5
2. Вихідні дані за варіантами	6
3. Теоретичні відомості	10
4. Приклад виконання завдання	17
5. Оформлення звіту до лабораторної роботи	22
Питання для самоперевірки знань	24
Список літератури	25

ВСТУП

Навчальна дисципліна «Інформаційні технології» є вибіркоким компонентом освітньо-професійної програми «Технологічні та логістичні системи у машинобудуванні» для підготовки здобувачів вищої освіти першого (бакалаврського) рівня за спеціальністю 131 «Прикладна механіка».

Метою викладання навчальної дисципліни є ознайомлення здобувачів з призначенням, можливостями та сучасним станом розвитку інформаційних технологій, формування знань, умінь та навичок ефективного застосування сучасних інформаційних технологій та програмного забезпечення для автоматизації інженерної діяльності, розв'язання різноманітних практичних задач за фахом.

У результаті вивчення навчальної дисципліни студент отримує відомості про архітектуру персонального комп'ютера IBM PC, принципи функціонування інформаційної системи та програмного забезпечення ЕОМ, основи роботи у середовищі об'єктно-орієнтованого та візуального програмування Delphi, сучасні засоби, прийоми і методи розробки програмного забезпечення, основні концепції та методологію об'єктно-орієнтованого програмування (ООП), принципи побудування графічних зображень, засоби обробки та аналізу інформації із застосуванням сучасних програмних засобів.

При виконанні лабораторної роботи «Розробка проекту Delphi» з навчальної дисципліни «Інформаційні технології» студенти набувають практичні навички розробки, налагодження та виконання в середовищі Delphi програм лінійної структури, що реалізують введення вихідних даних, обчислення та виведення результатів на формі Delphi.

Методичні вказівки до виконання лабораторної роботи містять формулювання завдання, вихідні дані за варіантами, теоретичні відомості за даною темою, докладний опис ходу виконання завдання на конкретному прикладі та правила оформлення звіту до лабораторної роботи. Також у документ включено список питань для самоперевірки знань студентів та список літератури, рекомендованої для вивчення.

МЕТА ВИКОНАННЯ РОБОТИ

Метою виконання лабораторної роботи на тему «Розробка проекту Delphi» є знайомство студентів із середовищем об'єктно-орієнтованого програмування Delphi, мовою програмування Object Pascal, технологіями візуального та подієвого програмування, особливостями створення стартової форми та програмного коду лінійної структури.

1. ФОРМУЛЮВАННЯ ЗАВДАННЯ

Вихідні дані за номерами варіантів наведено у розділі 2. Номер варіанта завдання відповідає номеру прізвища студента у журналі групи.

У середовищі Delphi необхідно розробити програму, що здійснює введення вихідних даних, обчислення за заданою формулою та виведення результатів розрахунку на формі Delphi. Рекомендованою версією середовища програмування є версія Delphi 7.

У програмі передбачити, щоб:

- поля введення вихідних даних набували лише числові значення;
- перехід від одного поля введення до іншого здійснювався при натисканні клавіші [Enter];
- поля виведення результатів розрахунку були недоступні для редагування.

2. ВИХІДНІ ДАНІ ЗА ВАРІАНТАМИ

Таблиця 2.1 – Вихідні дані відповідно до заданих варіантів

Номер варіанта	Завдання
1	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – довжини маятника l (м); – прискорення вільного падіння $g = 9.81$ (м/с²) <p>та виконує обчислення періоду коливання маятника</p> $t = 2\pi\sqrt{l/g} \text{ (с)}.$
2	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – мас двох тіл m_1 та m_2 (кг); – відстані між тілами r (м); <p>гравітаційної постійної $\gamma = 6.673 \cdot 10^{-11}$ (м³/(кг·с))</p> <p>та виконує обчислення сили тяжіння між тілами</p> $F = \frac{\gamma m_1 m_2}{r^2} \text{ (Н)}.$
3	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – швидкостей двох тіл v_1 та v_2 (м/с); – початкової відстані між тілами s (м) <p>та виконує обчислення часу, через який зустрінуться два тіла при рівномірному поступальному русі назустріч один одному,</p> $t = \frac{s}{v_1 + v_2} \text{ (с)}.$
4	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – об'ємів двох порцій води V_1 та V_2 (л); – їх температур t_1 та t_2 (°C) <p>та виконує обчислення об'єму та температури суміші порцій води</p> $V = V_1 + V_2 \text{ (л)}; \quad t = \frac{t_1 \cdot V_1 + t_2 \cdot V_2}{V_1 + V_2} \text{ (°C)}.$
5	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – висоти h (м); – прискорення вільного падіння $g = 9.81$ (м/с²) <p>та виконує обчислення швидкості та часу падіння каменю на поверхню землі з висоти h</p> $v = \sqrt{2gh} \text{ (м/с)}; \quad t = v/g \text{ (с)}.$
6	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – трьох опорів електричних гілок R_1, R_2 та R_3 (Ом) <p>та виконує обчислення повного опору ланцюга при їх паралельному з'єднанні</p> $R_{\text{пар}} = \frac{R_1 R_2 R_3}{R_1 R_2 + R_1 R_3 + R_2 R_3} \text{ (Ом)}.$

Продовження таблиці 2.1

Номер варіанта	Завдання
7	Розробити програму, яка здійснює введення – початкової швидкості рівноприскореного руху тіла v_0 (м/с); – прискорення тіла a (м/с ²); – часу руху тіла t (с) та виконує обчислення переміщення тіла за час t $s = v_0 t + \frac{at^2}{2} \text{ (м)}.$
8	Розробити програму, яка здійснює введення – маси тіла m (кг); – об'єму тіла V (м ³) та виконує обчислення щільності тіла $\rho = \frac{m}{V} \text{ (кг/м}^3\text{)}.$
9	Розробити програму, яка здійснює введення – електричного опору ділянки провідника R (Ом); – напруги на кінцях провідника U (В) та виконує обчислення сили струму, що тече у провіднику, $I = \frac{U}{R} \text{ (А)}.$
10	Розробити програму, яка здійснює введення – початкової швидкості руху тіла, кинутого вертикально нагору, v_0 (м/с); – прискорення вільного падіння $g = 9.81$ (м/с ²) та виконує обчислення максимальної висоти, на яку підніметься тіло, $h_m = \frac{v_0^2}{2g} \text{ (м)}.$
11	Розробити програму, яка здійснює введення – мас двох тіл m_1 та m_2 (кг); – швидкостей двох тіл v_1 та v_2 до зіткнення (м/с) та виконує обчислення швидкостей двох тіл після пружного зіткнення $v_1' = \frac{(m_1 - m_2)v_1 + 2m_2v_2}{m_1 + m_2} \text{ (м/с)}; \quad v_2' = \frac{(m_2 - m_1)v_2 + 2m_1v_1}{m_1 + m_2} \text{ (м/с)}.$
12	Розробити програму, яка здійснює введення – двох ємностей конденсаторів напруги C_1 та C_2 (мкФ) та виконує обчислення повної ємності ланцюга при їх послідовному з'єднанні $C_{\text{посл}} = \frac{C_1 C_2}{C_1 + C_2} \text{ (мкФ)}.$
13	Розробити програму, яка здійснює введення – температури повітря t (°С) та виконує обчислення швидкості звуку в повітрі при цій температурі $c = 331.6 \sqrt{1 + \frac{t}{273}} \text{ (м/с)}.$

Продовження таблиці 2.1

Номер варіанта	Завдання
----------------	----------

14	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – маси тіла m (кг); – кутовий швидкості обертального руху тіла ω (рад/с); – радіуса кола обертання r (м) <p>та виконує обчислення доцентрової сили, що діє на тіло при його обертальному русі,</p> $F_{\text{ц}} = m\omega^2 r \text{ (Н)}.$
15	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – напруги на кінцях провідника U (В); – сили струму, що тече у провіднику, I (А) <p>та виконує обчислення потужності електричного струму у провіднику</p> $P = UI \text{ (Вт)}.$
16	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – мас двох тіл m_1 та m_2 (кг); – швидкостей двох тіл v_1 та v_2 до зіткнення (м/с) <p>та виконує обчислення загальної швидкості обох тіл після непружного зіткнення</p> $v = \frac{m_1 v_1 + m_2 v_2}{m_1 + m_2} \text{ (м/с)}.$
17	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – висоти стовпа рідини в судині h (м); – прискорення вільного падіння $g = 9.81$ (м/с²) <p>та виконує обчислення швидкості витікання рідини з судини</p> $v = \sqrt{2gh} \text{ (м/с)}.$
18	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – маси тіла m (кг); – прискорення вільного падіння $g = 9.81$ (м/с²) <p>та виконує обчислення сили тяжіння, що діє на тіло,</p> $G = mg \text{ (Н)}.$
19	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – сили F (Н), що прикладена до тіла у його обертальному русі навколо деякої точки; – довжини перпендикуляра l (м), що опущений з центру обертання на лінію дії сили, <p>та виконує обчислення моменту сили щодо центру обертання</p> $M = Fl \text{ (Н}\cdot\text{м)}.$
20	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – лінійного переміщення тіла по траєкторії його рівномірного обертального руху s (м); – радіуса траєкторії обертання r (м) <p>та виконує обчислення кутового переміщення тіла</p> $\varphi = \frac{s}{r} \text{ (рад)}.$

Продовження таблиці 2.1

Номер варіанта	Завдання
----------------	----------

21	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – постійної сили F (Н), що переміщує тіло; – переміщення тіла s (м); – кута α (рад) між напрямками дії сили та переміщення <p>та виконує обчислення роботи, що здійснена силою,</p> $W = Fs \cos\alpha \text{ (Дж)}.$
22	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – сили струму в циліндричній котушці I (А); – числа витків котушки n; – довжини котушки l (м) <p>та виконує обчислення напруженості магнітного поля всередині котушки</p> $H = \frac{In}{l} \text{ (А/м)}.$
23	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – маси тіла m (кг); – прискорення вільного падіння $g = 9.81$ (м/с²); – висоти h (м), на яку піднімається тіло проти напрямку дії сили тяжіння, <p>та виконує обчислення роботи з підняття тіла</p> $W_{\text{п}} = mgh \text{ (Дж)}.$
24	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – площі перерізу труби A (м²); – швидкості перебігу рідини через переріз v (м/с); – часу t (с) <p>та виконує обчислення об'єму рідини, що протікає через переріз труби за цей час,</p> $V = Avt \text{ (м}^3\text{)}.$
25	<p>Розробити програму, яка здійснює введення</p> <ul style="list-style-type: none"> – часу обертання твердого тіла у його рівномірному обертальному русі t (с); – початкової кутової швидкості тіла ω_0 (рад/с); – кінцевої кутової швидкості тіла ω (рад/с) <p>та виконує обчислення кутового переміщення тіла</p> $\varphi = \frac{\omega_0 + \omega}{2} t \text{ (рад)}.$

3. ТЕОРЕТИЧНІ ВІДОМОСТІ

1. Особливості роботи у середовищі програмування Delphi.

Основні характеристики Delphi.

Середовище програмування Delphi є однією з провідних систем програмування, що використовуються для розробки сучасних прикладних програмних продуктів, і в першу чергу додатків Windows.

Систему Delphi відносять до категорії RAD-систем програмування (RAD – Rapid Application Development – швидка розробка додатків). В основі таких систем лежить технологія візуального проектування та подієвого програмування.

Система Delphi базується на використанні мови програмування Object Pascal, яка є логічним продовженням та розвитком класичної мови програмування Pascal. Середовище завантажується запуском файлу *delphi32.exe*. Програма, що знаходиться у Delphi на стадії розробки, називається *проект*ом.

Елементи екрану Delphi.

Екран середовища програмування Delphi (рис. 3.1) крім стандартних елементів вікна Windows включає:

- *палітру компонентів* – містить велику кількість вкладок із заготовками об'єктів додатку;
- *вікно форми* – є заготовкою вікна додатку, де розміщуються об'єкти;
- *вікно списку об'єктів* – містить список використовуваних об'єктів у вигляді деревоподібної структури;
- *вікно інспектора об'єктів* – містить вкладки для налаштування властивостей об'єктів та закріплення за об'єктами подій, на які вони мають реагувати;

- **вікно коду** – містить вкладку зі сформованим середовищем Delphi шаблоном програмного коду додатку, у якому користувач програмує у вигляді окремих процедур реакцію об'єктів на закріплені за ними події.

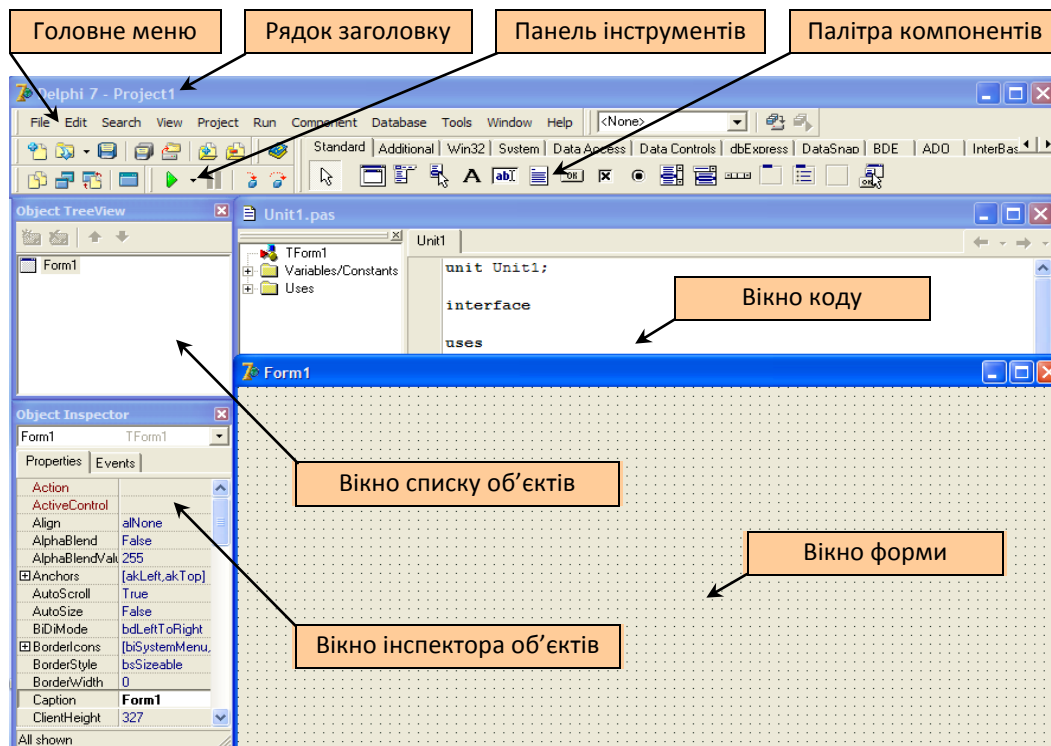


Рисунок 3.1 – Елементи екрану середовища програмування Delphi

Поняття об'єкта та події в Delphi.

Об'єкти додатку – це форма додатку та елементи, розташовані на ній. Кожен тип об'єктів має певний набір властивостей. Властивості об'єкта можуть змінюватися під час роботи додатку при впливі користувача на об'єкт. Такі дії, які вчиняє користувач щодо об'єкта, називаються **подіями**. Для кожного типу об'єктів існує певний набір подій, на які він може реагувати. Реакція об'єкта на закріплені за ним події програмується користувачем у вигляді окремих процедур (**обробників подій**).

2. Розробка першого проекту Delphi.

Створення стартової форми додатку Delphi, основні компоненти форми.

Під час розробки проекту Delphi система створює групу пов'язаних між собою програмних файлів. Тому рекомендується кожен проект, який створюється користувачем, зберігати в окремій папці.

Розробку проекту бажано розпочинати з його збереження. Попередньо створивши для проекту нову папку, необхідно викликати команду меню *File/Save as* для збереження файлу модуля форми *unit1.pas*, потім команду меню *File/Save Project as* для збереження головного файлу опису проекту *project1.dpr*. Інші пов'язані з проектом файли зберігаються автоматично.

Розробка проекту Delphi включає два взаємопов'язані компоненти:

- створення стартової форми додатку;
- створення програмного коду додатку.

Створення стартової форми додатку включає:

- перенесення з палітри компонентів на заготовку вікна додатку заготовок необхідних об'єктів додатку (компонентів форми); при цьому заготовки основних компонентів форми розміщені на вкладці *Standard (Стандартні)* палітри компонентів, ними є:
 - *Label (Напис)*;
 - *Edit (Текстове поле)*;
 - *Button (Командна кнопка)*;
 - та ін.;
- налаштування за допомогою вкладки *Properties (Властивості)* вікна інспектора об'єктів властивостей форми та властивостей об'єктів, розміщених на ній;
- закріплення за допомогою вкладки *Events (Події)* вікна інспектора об'єктів за об'єктами подій, на які вони мають реагувати.

Базовий набір подій Delphi.

Для кожного типу об'єктів існує певний набір подій, на які він може реагувати. У Delphi кожній події надано ім'я. **Базовий набір подій Delphi** включає такі події:

- **OnClick** – відбувається при клацанні кнопкою миші;
 - **OnDblClick** – відбувається при подвійному клацанні кнопкою миші;
 - **OnMouseMove** – відбувається при переміщенні миші;
 - **OnKeyPress** – відбувається при натисканні клавіші клавіатури;
 - **OnCreate** – відбувається при створенні об'єкта;
- та ін.

Поняття процедури обробки події.

Реакцією об'єкта на закріплену за ним подію має бути деяка дія. Ця дія програмується користувачем у вікні коду у вигляді окремої **процедури обробки події**. Під час запуску додатку ця процедура виконується у момент здійснення події над об'єктом.

Створення програмного коду додатку.

Середовище Delphi автоматично формує шаблон програмного коду додатку, додаючи до нього шаблони процедур обробки передбачених користувачем подій над об'єктами додатку. Тому для користувача **створення програмного коду додатку** полягає лише у доповненні шаблонів цих процедур.

3. Структура проекту Delphi.

Набір файлів, що становлять проект Delphi.

Проект Delphi є групою пов'язаних між собою програмних файлів, використовуючи які компілятор створює виконуваний код додатку (файл *project1.exe*). У найпростішому випадку проект складається з наступних файлів:

- головного файлу опису проекту *project1.dpr*;
- файлу модуля форми *unit1.pas*;
- файлу ресурсів *project1.res*;
- файлу опису форми *unit1.dfm*;
- та ін.

Призначення та структура головного файлу опису проекту.

Головний файл опису проекту project1.dpr містить інструкції, з яких починається виконання додатку. Він повністю формується середовищем Delphi і має такий вигляд:

```
program Project1;  
uses  
  Forms,  
  Unit1 in 'Unit1.pas' {Form1};  
{ $R *.res }  
begin  
  Application.Initialize;  
  Application.CreateForm (TForm1, Form1);  
  Application.Run;  
end.
```

У тексті файлу

- директива компіляції *{ \$R *.res }* підключає файл ресурсів *project1.res*, який містить ресурси додатку (піктограми, курсори, бітові образи та ін.);

- метод ***Initialize*** запускає ініціалізацію додатку;
- метод ***CreateForm*** ініціалізує форму додатку;
- метод ***Run*** запускає додаток на виконання.

Призначення та структура модуля форми проекту.

Модуль форми проекту unit1.pas містить опис форми додатку та процедур, що підтримують її роботу. Його шаблон формується середовищем Delphi при створенні користувачем нового проекту і має такий вигляд:

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms, Dialogs, StdCtrls;

type

TForm1 = class(TForm)

private

{ Private declarations }

public

{ Public declarations }

end;

var

Form1: TForm1;

implementation

{ \$R *.dfm }

end.

У тексті модуля

- у блоці ***type*** розділу ***interface*** описується тип-клас форми TForm1 на базі батьківського класу TForm;
- директива компіляції { \$R *.dfm } підключає файл опису форми ***unit1.dfm***, який формується середовищем Delphi на основі

зовнішнього вигляду форми і містить опис властивостей форми та властивостей об'єктів, розташованих на ній.

У процесі розробки користувачем проекту в блоці *type* розділу *interface* в опис типу-класу форми середовищем Delphi будуть додані описи об'єктів, розташованих на формі, і заголовки процедур обробки передбачених користувачем подій над об'єктами додатку, а в розділі *implementation* – шаблони цих процедур, що доповнюються користувачем.

4. Збереження проекту Delphi. Особливості компіляції та виконання програми.

Збереження та компіляція проекту Delphi.

Кожен новий проект рекомендується зберігати у попередньо створеній окремій папці. Для збереження проекту необхідно викликати команду меню *File/Save as* для збереження файлу модуля форми *unit1.pas*, потім команду меню *File/Save Project as* для збереження головного файлу опису проекту *project1.dpr*. Інші пов'язані з проектом файли зберігаються автоматично.

Для компіляції проекту після його збереження необхідно викликати команду меню *Project/Compile Project*. Процес компіляції складається з двох етапів. На першому етапі виконується перевірка тексту програми на відсутність помилок, на другому – генерується виконуваний код програми (файл *project1.exe*).

Повідомлення про помилки, попередження та підказки на етапі компіляції програми.

Процес та результат компіляції проекту відображаються у діалоговому вікні *Compiling*. У це вікно компілятор виводить кількість помилок, попереджень та підказок. Самі повідомлення про помилки, попередження та підказки знаходяться у нижній частині вікна редактора коду.

При виявленні у програмі *помилки (Errors)* на перший план виводиться вікно файлу, у якому знаходиться помилка, зазначена у першому повідомленні про помилки (зазвичай це файл *unit1.pas*). Після виправлення помилки виконується повторна компіляція.

При виявленні у програмі неточностей, які не є помилками, компілятор виводить *попередження (Warnings)* і *підказки (Hints)*.

Повідомлення про помилки часу виконання та винятки під час виконання програми.

Для *запуску проекту на виконання* після його компіляції необхідно завантажити файл *project1.exe* або у середовищі Delphi викликати команду меню *Run/Run*. Під час виконання програми можуть виникати *помилки часу виконання (run-time errors)* або *винятки (exceptions)*. Найчастіше їх причиною є неправильні вихідні дані.

4. ПРИКЛАД ВИКОНАННЯ ЗАВДАННЯ

Вихідні дані:

Розробити програму, яка здійснює введення двох чисел a і b та виконує обчислення їх суми $c = a + b$.

Хід виконання завдання:

1. У середовищі Delphi розробляємо програму, що здійснює введення вихідних даних, обчислення за заданою формулою та виведення результату розрахунку на формі Delphi:

- Налаштовуємо стартову форму – заготовку вікна програми (рис. 4.1).

Переносимо на форму зі вкладки *Standard (Стандартні)* палітри компонентів заготовки необхідних об'єктів додатку – *Label (Напис)*, *Edit (Текстове поле)* та *Button (Командна кнопка)*.

У вікні інспектора об'єктів на вкладці *Properties (Властивості)* змінюємо необхідні властивості форми та властивості об'єктів, розміщених на ній.

На етапі налаштування форми вводимо початкові значення чисел a і b , змінюючи значення властивостей *Text* для полей введення вихідних даних *Edit1* та *Edit2*.

Для того, щоб поле виведення результату розрахунку *Edit3* було недоступно для редагування, змінюємо для нього значення властивості *ReadOnly* з *False* на *True*.

У вікні інспектора об'єктів на вкладці *Events (Події)* за кнопками *Button1* («Скласти») та *Button2* («Вийти») закріплюємо події *OnClick*, на які вони мають реагувати виконанням обчислень та закінченням роботи програми.

За полями введення вихідних даних *Edit1* та *Edit2* закріплюємо події *OnKeyPress*. Реакцією на ці події повинен бути аналіз натиснутих клавіш клавіатури під час введення вихідних даних та дозвіл введення лише числових значень, а також перехід від одного поля введення до іншого при натисканні клавіші [Enter].

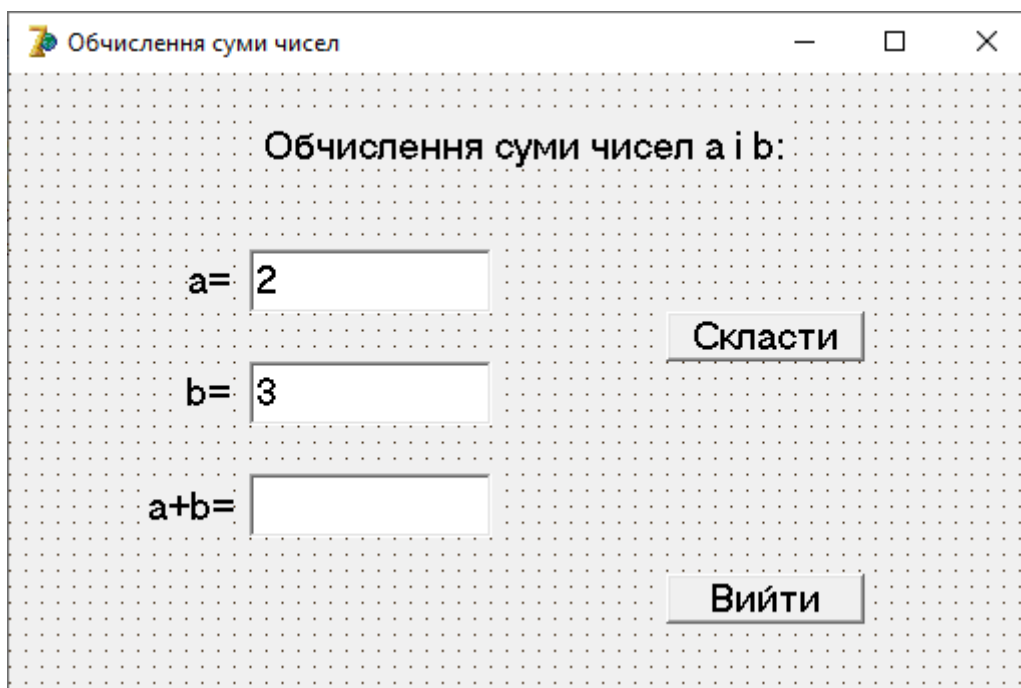


Рисунок 4.1 – Налаштування стартової форми програми

- Доповнюємо шаблон файла *Unit1.pas*.

Доповнений шаблон файлу *Unit1.pas* має наступний вигляд:

```
unit Unit1;  
  
interface  
  
uses  
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
    Dialogs, StdCtrls;  
  
type  
    TForm1 = class(TForm)  
        Label1: TLabel;  
        Label2: TLabel;  
        Label3: TLabel;  
        Label4: TLabel;  
        Edit1: TEdit;  
        Edit2: TEdit;  
        Edit3: TEdit;  
        Button1: TButton;  
        Button2: TButton;  
        procedure Button2Click(Sender: TObject);  
        procedure Button1Click(Sender: TObject);  
    private  
        { Private declarations }  
    public  
        { Public declarations }  
    end;  
  
var  
    Form1: TForm1;
```

implementation

```
{ $R *.dfm }
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
var a,b,c:real;
```

```
begin
```

```
  a:=StrToFloat(Edit1.Text);
```

```
  b:=StrToFloat(Edit2.Text);
```

```
  c:=a+b;
```

```
  Edit3.Text:=FloatToStr(c)
```

```
end;
```

```
procedure TForm1.Button2Click(Sender: TObject);
```

```
begin
```

```
  Close
```

```
end;
```

```
procedure TForm1.Edit1KeyPress(Sender: TObject; var Key: Char);
```

```
begin
```

```
  case Key of
```

```
    '0'..'9', '-', '.', #8 : ;
```

```
    #13 : Edit2.SetFocus;
```

```
    else Key:=chr(0)
```

```
  end;
```

```
end;
```

```
procedure TForm1.Edit2KeyPress(Sender: TObject; var Key: Char);
```

```
begin
```

```
  case Key of
```

```
    '0'..'9', '-', '.', #8 : ;
```

```
    else Key:=chr(0)
```

```
  end;
```

end;

end.

4. Запускаємо програму на виконання, вводимо вихідні дані та отримуємо результат роботи програми (рис. 4.2).

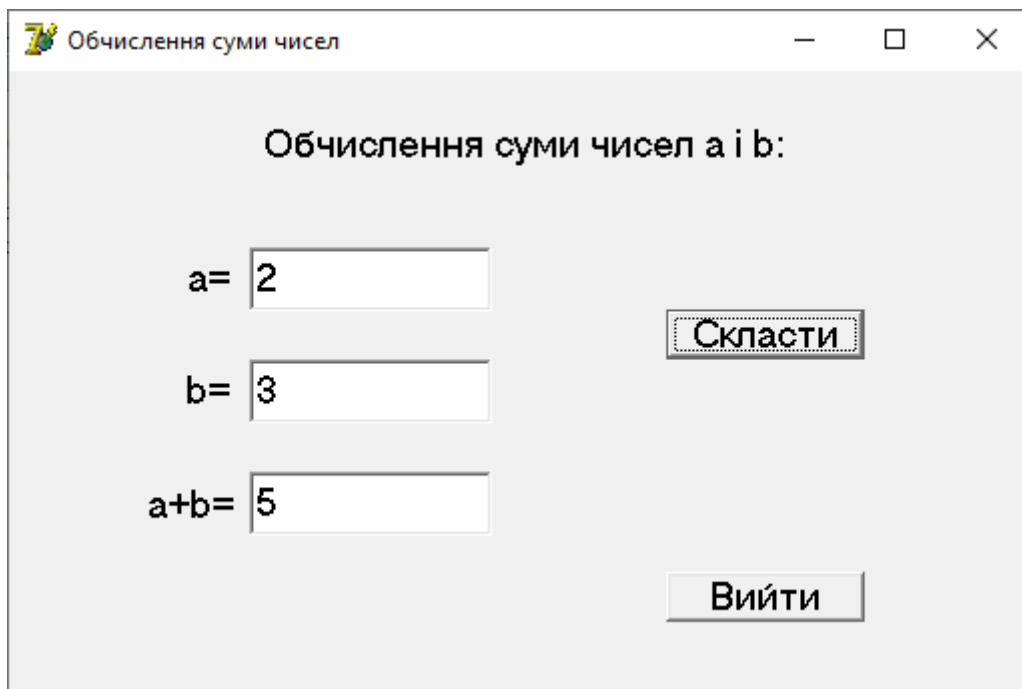


Рисунок 4.2 – Результат роботи програми побудови креслення квадрата та трикутника

5. ОФОРМЛЕННЯ ЗВІТУ ДО ЛАБОРАТОРНОЇ РОБОТИ

Звіт до лабораторної роботи необхідно оформити на аркушах формату А4.

Звіт повинен містити:

- титульний лист (приклад його оформлення показано на рис. 5.1);
- формулювання завдання (див. розділ 1);
- вихідні дані відповідно до заданого варіанту (див. розділ 2);
- хід виконання завдання (див. розділ 4):

1) текст розробленої програми обчислення за заданою формулою (вміст файлу *Unit1.pas*);

2) результат роботи програми (зображення вікна програми для введення вихідних даних та виведення результатів розрахунку).

Міністерство освіти та науки України
Національний технічний університет
«Харківський політехнічний інститут»

Лабораторна робота
з навчальної дисципліни «Інформаційні технології»
на тему «Розробка проекту Delphi»

Варіант 1

Виконав
студент гр. МІТ-223а
Іванов І.І.

Перевірила
доц., к.т.н. Третяк Т.Є.

Харків
2025

Рисунок 5.1 – Приклад оформлення титульного листа звіту
до лабораторної роботи

ПИТАННЯ ДЛЯ САМОПЕРЕВІРКИ ЗНАНЬ

1. Основні характеристики середовища програмування Delphi.
2. Елементи екрана середовища програмування Delphi.
3. Поняття об'єкта та події в середовищі програмування Delphi.
4. Створення стартової форми додатку Delphi, основні компоненти форми.
5. Базовий набір подій Delphi.
6. Поняття процедури обробки події в середовищі програмування Delphi.
7. Створення програмного коду додатка Delphi.
8. Сукупність модулів, що складають проект Delphi.
9. Призначення та структура головного модуля проекту Delphi.
10. Призначення та структура модуля форми проекту Delphi.
11. Особливості креслення на полотні Delphi ліній і контурів.
12. Особливості зафарбовування на полотні Delphi областей, обмежених контурами.
13. Вивід тексту на полотно Delphi.
14. Креслення прямої та ламаної ліній на полотні Delphi.
15. Креслення еліпса та окружності на полотні Delphi.
16. Креслення дуг еліпса та окружності на полотні Delphi.
17. Креслення секторів еліпса та окружності на полотні Delphi.
18. Креслення прямокутника або багатокутника на полотні Delphi.
19. Фарбування крапки на полотні Delphi.
20. Правила оголошення та використання в середовищі Delphi користувальницьких підпрограм.
21. Правила оголошення та використання в середовищі Delphi користувальницьких модулів.
22. Особливості взаємодії основної та додаткової форм проекту Delphi.
23. Поняття класу, об'єкта та метода Delphi.

СПИСОК ЛІТЕРАТУРИ

1. Ковалюк Т. В. Основи програмування: підручник. – К.: Видавнича група ВНУ, 2005. – 384 с.
2. Шаховська Н. Б., Голощук Р. О. Алгоритми і структури даних: посіб. – Львів: Магнолія, 2010. – 215 с.
3. Миронченко А. С. Імперативне та об'єктно-орієнтоване програмування на Turbo Pascal та Delphi. Глибоке занурення. – Одеса: ВМВ, 2007. – 408 с.
4. Безменов М. І. Основи програмування в середовищі Delphi: навч. посіб. – Харків: НТУ «ХП», 2010. – 608 с.
5. Кащев Л. Б., Коваленко С. В., Коваленко С. М. Інформатика. Основи візуального програмування [Електронний ресурс]: навч. посіб. – Харків: Веста, 2011. – 192 с.
6. Яременко Г. І., Копил Д. В. Програмування в середовищі Delphi: навч. посіб. – Черкаси: ЧДТУ, 2011. – 188 с.
7. Методичні вказівки до виконання Розрахунково-графічного завдання з навчальної дисципліни «Інформаційні технології» для студентів усіх форм навчання за спеціальністю 131 «Прикладна механіка» / уклад.: Т. Є. Третяк, О. Л. Мироненко, С. О. Мироненко. – Харків: НТУ «ХП», 2024. – 38 с.

Навчальне видання

Методичні вказівки
до виконання лабораторної роботи «Розробка проекту Delphi»
з навчальної дисципліни «Інформаційні технології»
для студентів усіх форм навчання
за спеціальністю 131 «Прикладна механіка»

Укладачі:

ТРЕТЯК Тетяна Євгенівна
МИРОНЕНКО Олександр Леонідович
МИРОНЕНКО Сергій Олександрович

Відповідальний за випуск Третяк Т. Є.
Роботу до видання рекомендував проф. Клочко О. О.

В авторській редакції

План 2025 р., поз. 267.

Підп. до друку 10.03.2025 р. Гарнітура Times New Roman. Ум. друк. арк.
1,2.

Видавничий центр НТУ «ХПІ», вул. Кирпичова, 2, м. Харків, 61002
Свідоцтво про державну реєстрацію ДК № 5478 від 21.08.2017 р.

Електронне видання