

Лабораторна робота 3.

Розробка простого крос-платформного застосування засобами Mono

Мета роботи: Розглянути засоби програмної платформи Mono для створення крос-платформних застосувань.

Підготовка необхідного програмного забезпечення

1. Завантажити з сайту розробника [1; 2; 3] та встановити в ОС Windows програмну платформу Mono. (Далі використовуватиметься версія Mono-3.0.6). Додати шлях для *Mono-3.0.6\bin* до змінних середовища Windows (напр. *D:\Program Files\Mono-3.0.6*).
2. Встановити Mono та MonoDevelop в ОС Linux (далі викорисовуватиметься Ubuntu 10.04). Для встановлення MonoDevelop та інших компонентів можна скористатись Synaptic Package Manager'ом.

Приклад:

Засобами Visual Studio створити бібліотеку класів .Net з іменем *ClassLibrary1.dll* для обчислення суми натуральних чисел від 1 до N включно.

Засобами Mono в ОС Windows створити консольну програму, яка запитує у користувача натуральне число N та, з використанням *ClassLibrary1.dll*, обчислює та повідомляє суму всіх натуральних чисел від 1 до N включно.

Пересвідчитись у працездатності консольного Mono-застосунку в ОС Windows та Linux.
В ОС Linux створити віконну програму з такою ж, як і у консольної, функціональністю.
Пересвідчитись у працездатності віконного Mono-застосунку в ОС Windows та Linux.

Порядок виконання роботи

Консольний проект

1. Бібліотека класів.

Засобами Visual Studio створити збірку *ClassLibrary1.dll*, з таким початковим кодом:

```
namespace ClassLibrary1
{
    public class Summing
    {
        public int Summ(int n) {
            int s = 0;
            for (int i = 1; i <= n; i++) s += i;
            return s;
        }
    }
}
```

2. Консольне Mono-застосування в ОС Windows.

2.1. У вибраному для роботи каталозі (наприклад *D:\Program Files\Mono-3.0.6\wrk\sum*) створити файл *sm.cs* з таким змістом:

```
using System;
using ClassLibrary1;

class Program
{
    static void Main(string[] args)
    {
        Console.WriteLine("Type the number");
        string s = Console.ReadLine();
        int N;
        Summing sm = new Summing();
        while (!s.Equals(""))
        {
            N = Int32.Parse(s);
```

```

        Console.WriteLine("sum(i,i=0..N)=" + sm.Summ(N));
        Console.WriteLine("Type the number or ENTER");
        s = Console.ReadLine();
    }
}
}

```

2.2. Розмістити у тому ж каталозі копію *ClassLibrary1.dll*

2.3. Запустити *Mono-3.0.6 Command Prompt*.

2.3.1. Перейти в робочий каталог командою *cd /D D:\Program Files\Mono-3.0.6\wrk\sum*

2.3.2. Скомпілювати програму з підключенням (reference) бібліотеки класів:

mcs sm.cs /r:ClassLibrary1.dll

2.3.3. Запустити програму в *Mono-3.0.6 Command Promp*:

mono sm.exe

або *sm.exe*

або іншим способом у середовищі Windows (напр. подвійним кліком)

```

D:\Program Files\Mono-3.0.6\wrk\sum>mcs sm.cs /r:ClassLibrary1.dll
D:\Program Files\Mono-3.0.6\wrk\sum>mono sm.exe
Type the number
5
sum(i,i=0..N)=15
Type the number or ENTER
10
sum(i,i=0..N)=55
Type the number or ENTER

```

2.1. Скопіювати *sm.exe* та *ClassLibrary1.dll* в середовище Ubuntu, наприклад в */home/sum*

2.2. В командному вікні (Terminal):

Перейти в каталог *sum*;

Запустити *sm.exe* на виконання командою *pubuntu@pubuntu:~/sum\$ mono sm.exe*

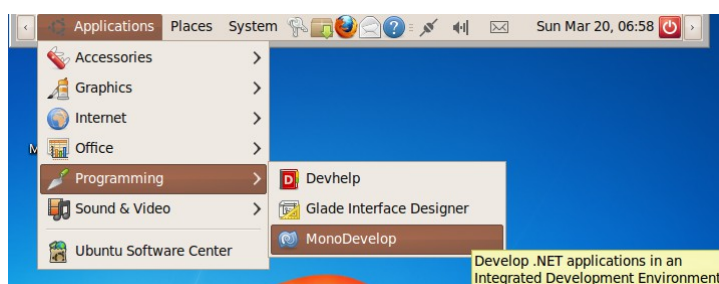
```

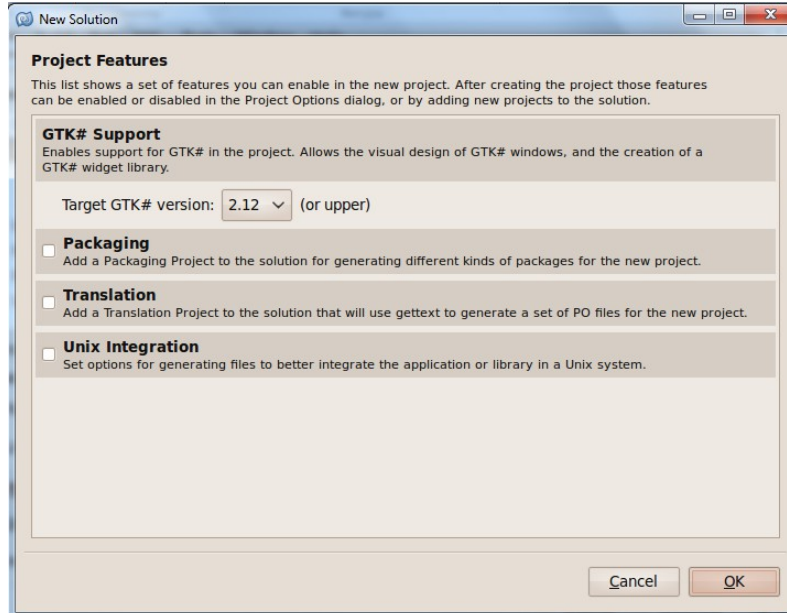
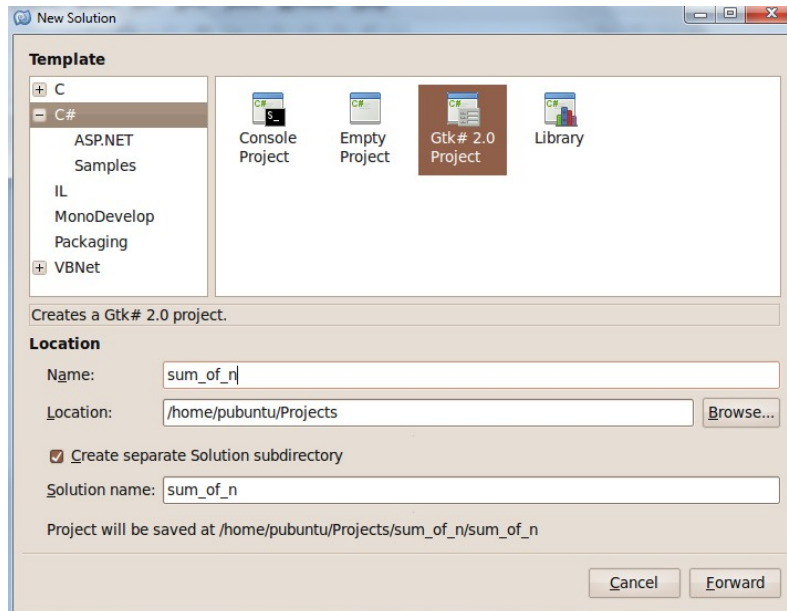
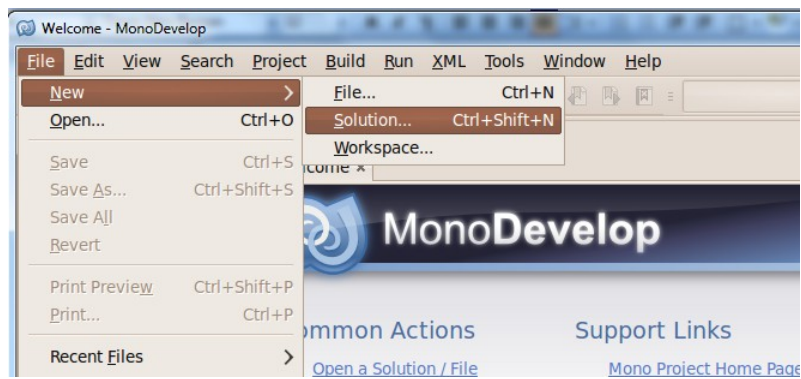
pubuntu@pubuntu: ~/sum
File Edit View Terminal Help
pubuntu@pubuntu:~$ cd sum
pubuntu@pubuntu:~/sum$ ls
ClassLibrary1.dll sm.cs sm.exe
pubuntu@pubuntu:~/sum$ mono sm.exe
WARNING: The runtime version supported by this application is unavailable.
Using default runtime: v1.1.4322
Type the number
5
sum(i,i=0..N)=15
Type the number or ENTER
11
sum(i,i=0..N)=66
Type the number or ENTER
pubuntu@pubuntu:~/sum$

```

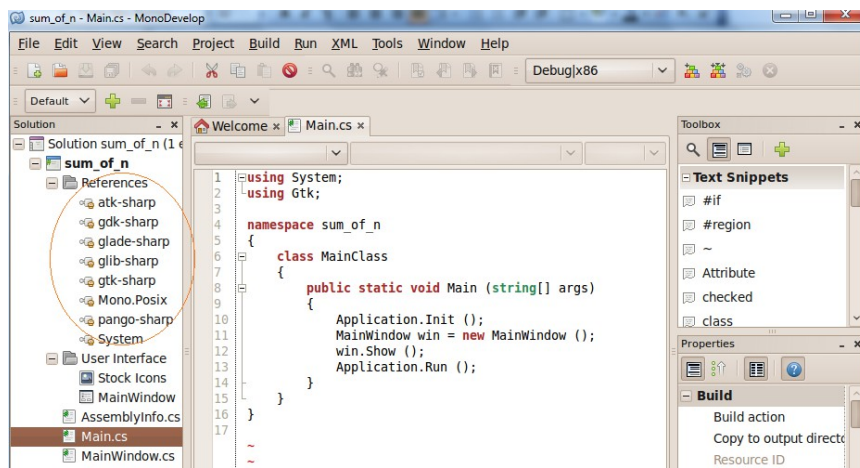
3. Віконне Моно-застосування в ОС Linux.

3.1. В середовищі Ubuntu запустити MonoDevelop, створити новий проект типу Gtk# 2.0 Project (див. скріншоти).

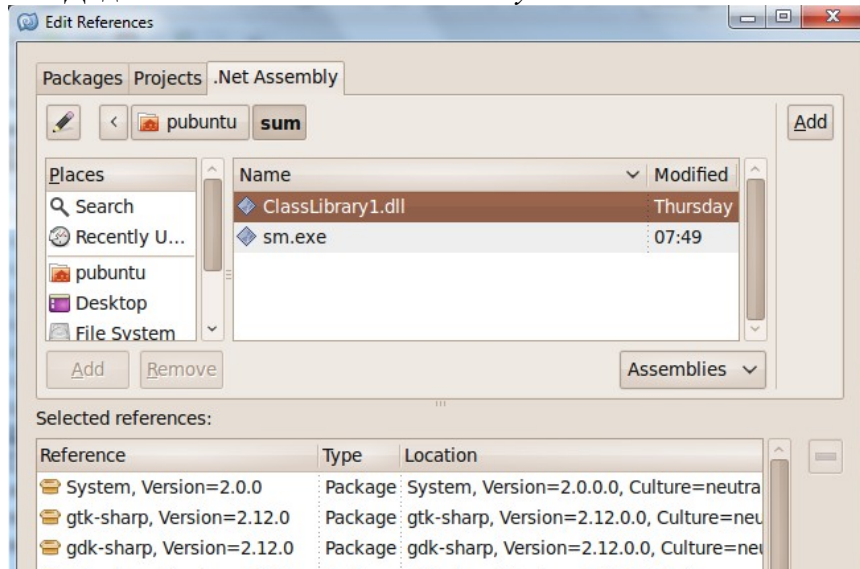




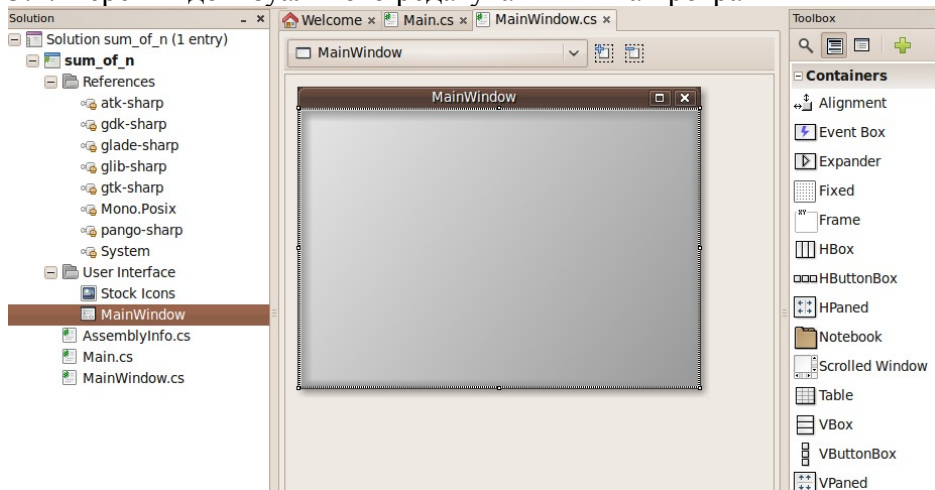
3.2. В References всі збірки мають бути доступні. Якщо це не так – встановити необхідні компоненти через Synaptic Package Manager.



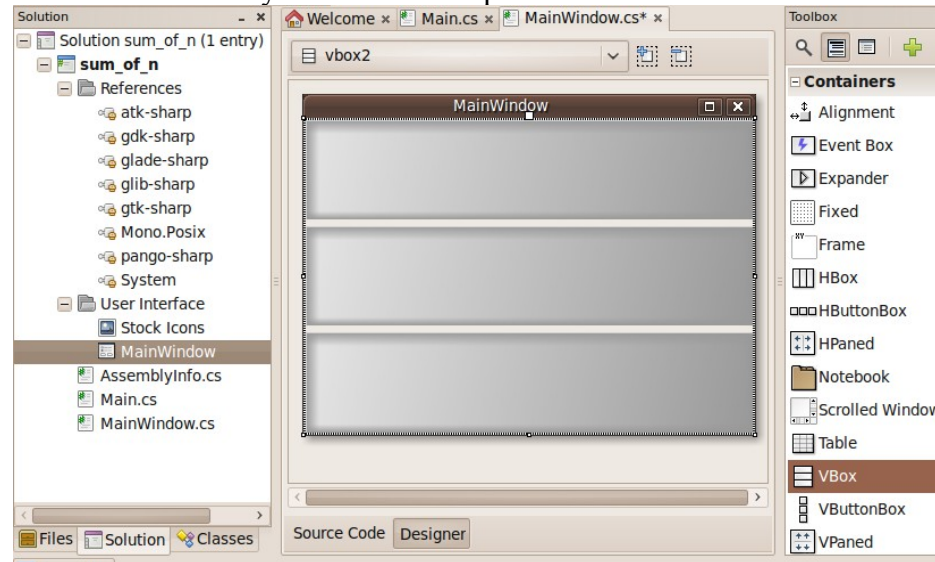
3.3. Додати посилання на *ClassLibrary1.dll*



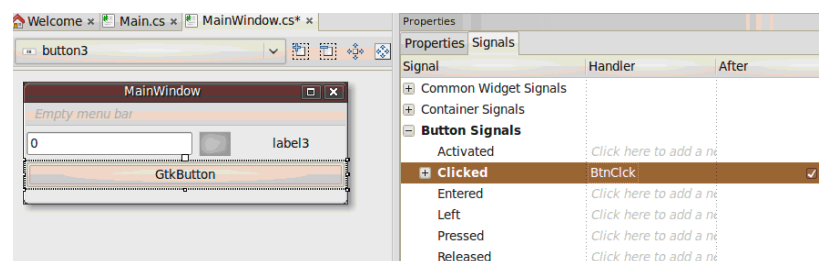
3.4. Перейти до візуального редагування вікна програми



3.4.1. Розмістити у вікні контейнер VBox



3.4.2. У контейнері VBox розмістити а) Menu Bar; б) HBar (в ньому Entry та Label); в) Button. У вікні властивостей кнопки визначити для сигналу Clicked обробку методом BtnClick.

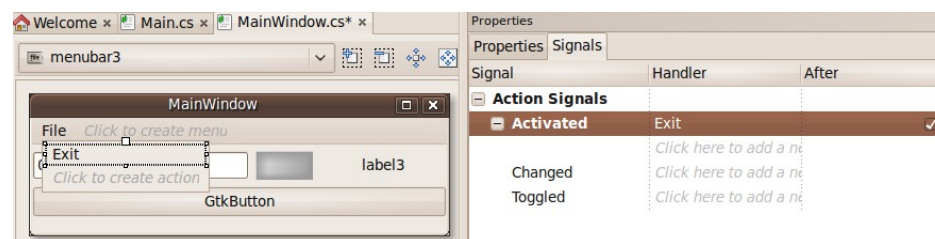


3.4.3. Шаблон обробника доповнити кодом

protected virtual void BtnClick (object sender, System.EventArgs e)

```
{
    Summing sm = new Summing();
    label3.Text=sm.Summ(int.Parse(entry3.Text)).ToString();
}
```

3.4.4. Додати пункт меню Exit та передбачити його обробку



3.4.5. Повний код MainWindow.cs:

```
using System;
using Gtk;
using ClassLibrary1;

public partial class MainWindow : Gtk.Window
{
    public MainWindow () : base(Gtk.WindowType.Toplevel)
    {
        Build ();
    }
    protected void OnDeleteEvent (object sender, DeleteEventArgs a)
    {
    }
```



```

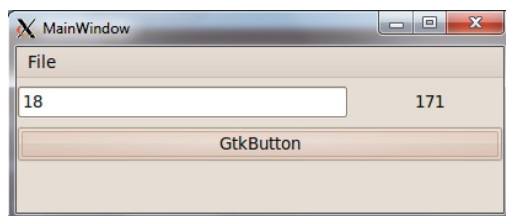
        Application.Quit ();
        a.RetVal = true;
    }

    protected virtual void BtnClick (object sender, System.EventArgs e)
    {
        Summing sm = new Summing();
        label3.Text=sm.Summ(int.Parse(entry3.Text)).ToString();
    }

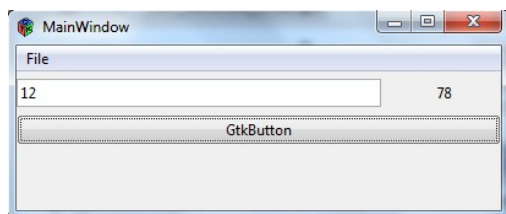
    protected virtual void Exit (object sender, System.EventArgs e)
    {
        Application.Quit();
    }
}

```

3.4.6. Скомпілювати та запустити проект у MonoDevelop, або у Терміналі
pubuntu@pubuntu:~\$ cd Projects/sum_of_n/sum_of_n/bin/Debug
pubuntu@pubuntu:~/Projects/sum_of_n/sum_of_n/bin/Debug\$ mono sum_of_n.exe



3.4.6. Скопіювати файли проекту у середовище Windows *sum_of_n.exe*, *sum_of_n.exe.mdb* та *ClassLibrary.dll* запустити з командного рядка *Mono-3.0.6 Command Prompt* командою:
mono sum_of_n.exe



Завдання для самостійного виконання

1. Розробити консольне та віконне застосування з функціональністю у відповідності до варіанта завдання (з використанням різноманітних віджетів у віконн. заст.). Всі обчислення виконуються за допомогою окремої збірки dll.

№ варіанта	Зміст завдання
1	Користувач вводить рядок. Програма визначає чи містить воно повтори, (підрядок принаймні з двох літер, який зустрічається більше одного разу). Наприклад рядок abbak – не містить повторів, а abbbab – містить два повтори підрядка ab та два повтори підрядка bb.
2	Користувач вводить рядок. Програма знаходить його довжину K та визначає, чи є воно (і яким по порядку) простим числом.
3	Користувач вводить рядок та число K. Програма знаходить кількість N слів у рядку та визначає найбільший спільний дільник чисел N та K.

4	Користувач вводить рядок. Програма знаходить кількість літер N1 та пробілів N2 у ньому, знаходить суму чисел від $\min(N1, N2)$ до $\max(N1, N2)$ включно.
5	Користувач вводить рядок та число M. Програма знаходить довжину N найбільшого слова у рядку, визначає найменше спільне кратне чисел M і N.
6	Користувач вводить рядок. Програма визначає чи є воно, або будь-який його префікс паліндромом (напр. рядок abbak – не паліндром, але його префікс abba – паліндром).
7	Користувач вводить два рядки. Програма визначає їх найбільший спільний префікс (напр. найбільший спільний префікс у abcs та abfgr є ab. А у abcs та fghrt – порожній рядок).
8	Програма визначає чи є в тексті записи цілих та/або дійсних чисел.
9	Користувач вводить рядок. Програма визначає довжину кожного слова тексту.
10	Користувач вводить слово. Програма повертає всі його анаграми (всі перестановки літер).
11	Програма отримує два рядки та змішує їх (напр. для abc та FGHK повертає aFbGcHK).
12	Програма повідомляє кількість одиниць у двійковому представленні числа та саме двійкове представлення.
13	Програма повертає кількість та список парних цифр у отриманому від користувача числі.
14	Користувач вводить два рядки. Програма визначає їх найбільший спільний суфікс. Наприклад найбільший спільний суфікс у abcsdf та abfgrsdf є sdf. А у abcs та fghrt – порожній рядок.
15	Рядки Фібоначчі обчислюють так: $f_0 = b$, $f_1 = a$, $f_n = f_{n-1} + f_{n-2}$, $n \geq 2$, де знаком + позначена операція конкатенації. Наприклад $f_2 = ab$, $f_3 = aba$, $f_4 = abaab$ тощо. Програма знаходить перші N рядків Фібоначчі та довжину кожного з них.
16	Програма перевіряє, чи є N-значний номер квитка "щасливим". Програма повертає висновок та суму перших і останніх N/2 цифр ("щасливим" вважається номер, в якому сума перших N/2 цифр збігається з сумою останніх N/2 цифр).
17	Користувач вводить рядок та число N. Програма виконує циклічний зсув рядка вправо на N слів.

Питання для самоконтролю

1. Програмна платформа Mono. Призначення, функціонування, застосування.
2. Програмна платформа Mono та .Net.
3. Інструментарій/компоненти Mono.
4. IDE для роботи з Mono.
5. Створення консольного/віконного застосування в Mono.
6. Використання застосунків Mono.

Посилання

1. <http://www.mono-project.com/>
2. <http://www.mono-project.com/download/>
3. <http://www.mono-project.com/docs/getting-started/install/windows/>
4. <http://www.mono-project.com/docs/gui/gtksharp/>
5. https://www.ibm.com/developerworks/ru/library/os_mono1/
6. https://www.ibm.com/developerworks/ru/library/l-Mono_2/