

ISSN 2307-9851

НАУКОВО-МЕТОДИЧНИЙ
ЖУРНАЛ

Виходить 8 разів на рік

Видається з лютого 1998 року

Засновники:

Інститут педагогіки НАПН України,
Інститут інформаційних технологій
і засобів навчання НАПН України,
Редакція журналу

Журнал видається за сприяння
Міністерства освіти і науки України
Свідоцтво про реєстрацію
серія КВ №12217-1101ПР
від 17.01.2007

Передплатний індекс 74248

Журнал включено до Переліку
наукових фахових видань України
у галузі педагогічних наук,
Наказ МОН України
від 29.09.2014 року №1081

Журнал внесений до
наукометричної бази даних РИНЦ

Затверджено Вченою радою
Інституту педагогіки НАПН України,
протокол №6 від 31 травня 2018 р.

Головний редактор
ЛАПІНСЬКИЙ В.В.

Заступник головного редактора
КАЛІНІНА Л.М.

Редактор
ВОВКОВІНСЬКА Н.В.

E-mail: csf22101@ukr.net

Офіційний сайт журналу:
www.csf221.wordpress.com

КОМП'ЮТЕР у школі та сім'ї

№3 (147) ♦ 2018

ЗМІСТ

ВИЩА ОСВІТА

Івлієва О. М. Можливості застосування MS EXCEL у процесі вивчення математичних дисциплін студентами-інформатиками _____ **3**

Чхало О. М. Застосування технології BYOD в освітньому процесі аналітичної хімії _____ **10**

ПИТАННЯ ТЕОРІЇ

Рудик О. Б. Налаштування взаємодії WXWIDGETS і CODEBLOCKS для забезпечення осучаснення навчання інформатики _____ **16**

Калініна Л. М., Топузов М. О. Аналіз базових засад управління опорним закладом освітнього округу _____ **21**

ІКТ В ОСВІТІ

Міщенко С. М. Автоматизована універсальна тестова система діагностики якості освіти «ТЕСТОРИУМ» _____ **27**

ПЕДАГОГІЧНИЙ ДОСВІД

Пількевич Л. З досвіду роботи психологічної служби у «Дитячій інженерній академії» _____ **32**

НОРМАТИВНІ ДОКУМЕНТИ

Інформатика. Навчальна програма вибірково-обов'язкового предмету для учнів 10 -11 класів загальноосвітніх навчальних закладів (рівень стандарту) _____ **37**

На першій сторінці обкладинки:

«День вишиванки» в Інституті педагогіки НАПН України

На другій сторінці обкладинки:

*Засідання Президії НАПН України, розгляд питання:
«Про результати дослідження в Інституті педагогіки
НАПН України проблем науково-методичного забезпечення
природничо-математичної освіти в старшій школі
(2015-2018 рр.)».*

УДК 37.01:004.4'2

НАЛАШТУВАННЯ ВЗАЄМОДІЇ wxWIDGETS I CODEBLOCKS ДЛЯ ЗАБЕЗПЕЧЕННЯ ОСУЧАСНЕННЯ НАВЧАННЯ ІНФОРМАТИКИ

Рудик Олександр Борисович

кандидат фізико-математичних наук, доцент Інституту післядипломної педагогічної освіти Київського університету імені Бориса Грінченка, rudykob@gmail.com



Анотація. Статтю присвячено аналізу й вирішенню проблем, пов'язаних зі зміною парадигми вивчення інформатики у школі: істотним зростанням кількості навчальних годин на вивчення програмування. Обґрунтовано вибір програмного забезпечення для навчального процесу, подано конкретні рекомендації щодо його встановлення та описано можливі проблеми такого встановлення.

Ключові слова: подійно та об'єктно орієнтоване програмування, C++, CodeBlocks, wxWidgets.

Затвердження навчальної програми курсу «Інформатика, 5–9 класи загальноосвітніх навчальних закладів (для учнів, які вивчали інформатику в 2-4 класах)» наказом Міністерства освіти і науки України від 02.02.2016 № 73 породило низку таких поки що нерозв'язаних проблем:

- програмування здебільшого вже не процедурно орієнтоване, до якого звикли вчителі старшого віку, а є подійно та об'єктно орієнтованим;

- попередні спроби навчати процедурно орієнтованого програмування були невдалими у більшості закладів середньої освіти;

- багато вчителів не мають не лише досвіду навчання (хоча б процедурно орієнтованому програмуванню), але навіть знань і навичок, достатніх для кодування алгоритмів у сучасних середовищах програмування;

- для значної частини вчителів і учнів характерним є низький рівень компетентностей щодо конструювання й запису алгоритмів;

- відсутність найближчим часом безоплатних для учнів підручників і посібників (наслідок положень чинних нормативних актів та вже проведених тендерів);

- намагання авторів підручників використовувати лише навчальні середовища програмування (наприклад, Lazarus), що істотно знижує мотивацію навчання;

- відсутність україно- і російськомовних джерел щодо важливих і важких для сприйняття аспектів вивчення актуальних для індустрії програмного забезпечення мов і середовищ програмування;

- відсутність для багатьох програмних засобів (систем програмування), задіяних в індустрії програмного забезпечення, стислого викладу, прийнятного для навчального процесу, навіть у англійськомовних джерелах;

- майже всі джерела щодо актуальних в індустрії програмного забезпечення мов і середовищ програмування розраховані на програмістів із досвідом або хоча б на старшокласників. Копіювання їх (лише з перекладом) у навчальних матеріалах для учнів

нічим не виправдане. Вони вимагають тривалої і кропіткої роботи щодо пристосування їх до рівня школярів. У першу чергу щодо логіки і стислості викладу змісту, зрозумілості учнями;

- як зауважено на семінарі методистів ОППО 21.06.2018 у Києві, згідно з чинною програмою, учні повинні уміти порівнювати роботу у різних середовищах програмування. Відповідне зауваження (нагадування) вирішили у 2018 році не включати до методичних рекомендацій МОН України, але зробити це до наступного навчального року. Причина цього така: у 2018/2019 навчальному році не буде 11-класників, яких навчатимуть за новою програмою, тому вимагати досягнення цієї компетенції вже цього року некоректно.

Окрім цього, однією з основних проблем є проблема вибору програмного забезпечення. Нині її вирішення повністю покладено на вчителя. Для усвідомленого вибору програмного забезпечення, яке передбачається використовувати в освітньому процесі, вчителю потрібно детально ознайомитися з можливими його варіантами.

Остання з достатньо обґрунтованих і відомих вітчизняних робіт щодо вибору мови й системи програмування для використання в освітньому процесі опублікована у 2014 році [4]. Отримані автором висновки базуються на матеріалах анкетування учасників (учнів і тренерів) четвертого етапу олімпіад з інформатики, що відбувалися у 2010 – 2013 роках і адекватно характеризували стан проблеми на той час.

П'ять років, що минули, для галузі, в якій ми працюємо, є надзвичайно великим терміном, протягом якого виникло багато нового, але незмінними залишилися вимоги переходу від варіантів процедурного програмування до програмування подійно й об'єктно орієнтованого. Тому нині основною задачею, яку необхідно вирішувати, є задача реального упровадження подійно й об'єктно орієнтованого програмування з переважним використанням систем візуальної підтримки розроблення програм.

Найпривабливішим варіантом вибору програмного забезпечення (щонайменше, для класів з поглибленим вивченням математики й інформатики) є поєд-

нання CodeBlock, C++ і wxWidgets. Чому? На це є щонайменше такі чотири причини.

Поєднання CodeBlock, C++ і wxWidgets є багато платформним і вільно поширюваним. Останнє істотно для українських загальноосвітніх навчальних закладів, у яких платне програмне забезпечення на законних підставах може з'явитися лише як подарунок сторонніх осіб (згідно зі статтями законодавства щодо захисту авторських прав — чинні програми не вимагають використання пропрієтарного програмного забезпечення).

Як свідчить статистика результатів IV етапу Всеукраїнської учнівської олімпіади з інформатики й відбірково-тренувальних до Міжнародної олімпіади, поєд-

нання CodeBlock і C++ є характерною ознакою інструментарію учасників олімпіади, що отримали найкращі результати.

Мову C++ широко використовують у сучасній індустрії програмного забезпечення.

Бібліотека wxWidgets повноцінно реалізує концепції подійно та об'єктно орієнтованого програмування для створення застосунків з повноцінним графічним інтерфейсом.

У початковому вигляді вікна CodeBlocks (одразу після створення нового проекту wxWidgets) користувачі Lazarus'у одразу впізнають знайомі елементи інтерфейсу (рис. 1), а у процесі роботи вони зіткнуться зі схожими алгоритмами роботи.

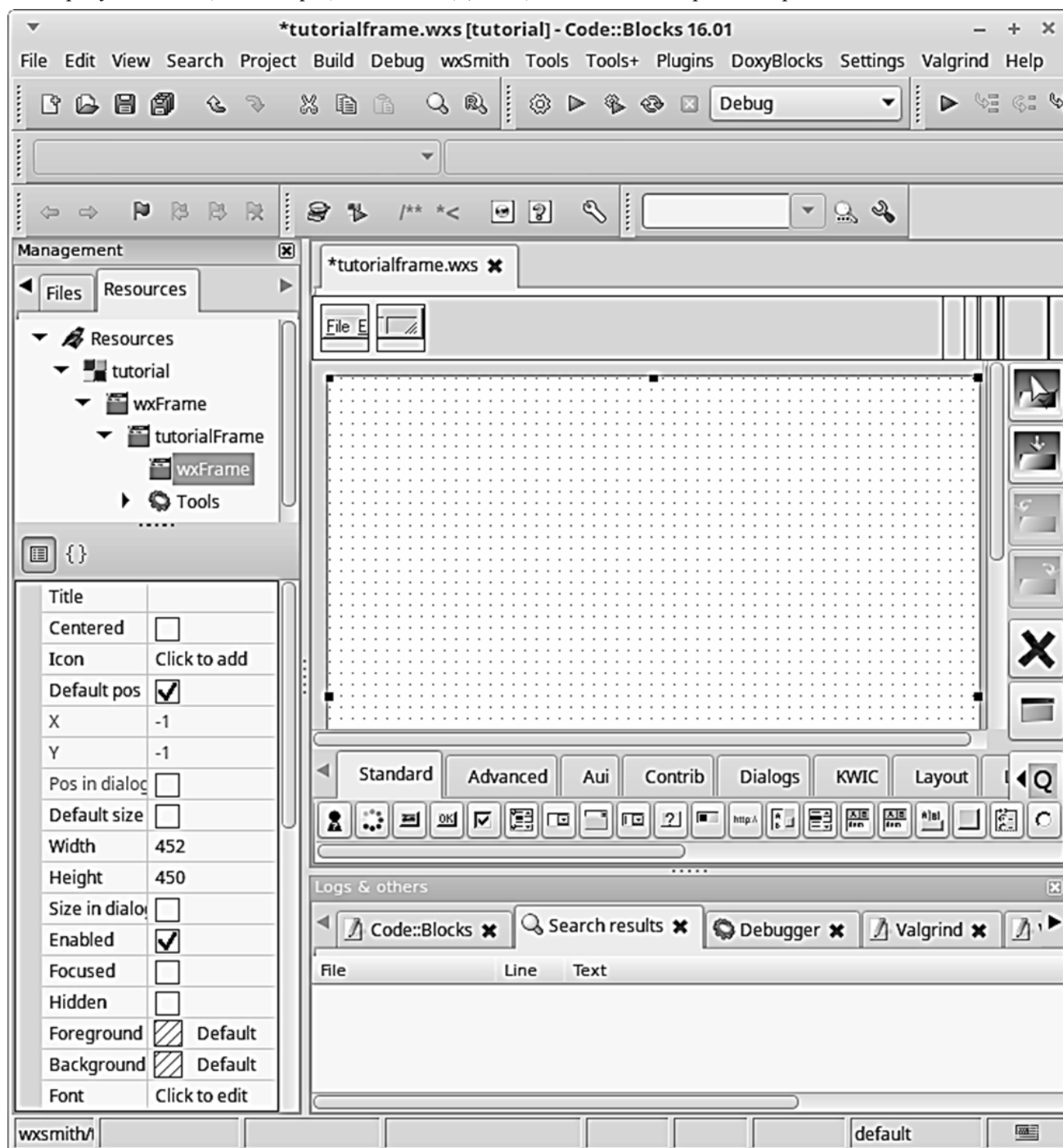


Рис. 1. Вигляд вікна CodeBlocks після створення нового проекту wxWidgets

Примітка. Для wxWidgets (у поєднанні з C++) є альтернатива — Qt. Ця альтернатива також багато платформна. Її повна версія вимагає оплати. Є безкоштовний урізаний варіант. Але не ці міркування зумовили вибір автора спочатку запропонувати саме wxWidgets. Щонайменше зараз приклади кодів та їхні описи для Qt видаються громіздкими й важчими для сприйняття, ніж приклади для wxWidgets.

Алгоритм встановлення програмного забезпечення для ОС Windows 10

(згідно з рекомендаціями [1])

1. Встановити CodeBlocks:

- перейти на сайт codeblocks.org;
- у розділі *Downloads* обрати *Download the binary release*;
- обрати пакет залежно від платформи, але з комп'лятором (mingw у назві). Наприклад, для ОС Windows 7, 8, 10 — `codeblocks-16.01mingw-setup.exe`;

- обирати джерело завантаження (довільне з наявних);

- знайти завантажений пакунок у теці *Завантаження* і запустити його на виконання, попередньо закривши вікна інших застосунків;

- далі діяти традиційно.

2. Встановити WxWidgets:

- перейти на сайт wxwidgets.org;
- у меню стартової сторінки сайту обрати розділ *Downloads* (Завантаження) або скористатися кнопкою *Download now* (Завантажити зараз) на головній сторінці сайту;

- у переліку запропонованих посилань обрати *Latest Stable Release* (Остання стабільна версія) Windows ZIP;

- перейти у теку *Завантаження* і обрати щойно завантажений файл;

- розпакувати архів (бажано у той самий розділ, у якому вже встановлено *CodeBlocks*);

3. Додати (через ;) значення змінної середовища Path адресою теки bin із засобами MinGW (наприклад, D:\CodeBlocks\MinGW\bin).

4. Налаштувати використання WxWidgets у середовищі CodeBlocks:

- викликати командний рядок:
 - або в рядку пошуку ввести `cmd` та натиснути клавішу `Enter`;
 - або натиснути клавіші `Windows + R`, у вікні, що з'явиться ввести `cmd` та натиснути `ОК`;
- перейти у розділ, у якому встановлено wxWidgets (наприклад, D:);

- змінити поточну теку на `wxWidgets\build\msw` (щоб не помилитися, використати засоби копіювання та вставлення);

- виконати вказівку побудови wxWidgets (усе одним рядком):

```
mingw32-make -f makefile.gcc BUILD=release
SHARED=1 MONOLITHIC=1 UNICODE=1 CXXFLAGS=-std=gnu++11
```

Виконання цієї вказівки може тривати від 15 хвилин до декількох годин (для малопотужних комп'ютерів). По завершенню побудови wxWidgets закрити вікно рядка вказівок.

Примітка. Кроки 3 і 4 можна виконати, використавши такий `bat`-файл:

```
SET PATH=%PATH%;"C:\ProgramFiles (x86)
\CodeBlocks\MinGW\bin"
```

d:

```
cd d:\wxWidgets\build\msw
```

```
mingw32-make -f makefile.gcc BUILD=release
SHARED=1 MONOLITHIC=1 UNICODE=1 CXXFLAGS=-std=gnu++11
```

Для поданого прикладу:

- `C:\Program Files (x86)\CodeBlocks\MinGW\bin` — (повна) адреса теки розташування засобів MinGW, у тому числі програми `mingw32-make`;

- `d:\wxWidgets` — (повна) адреса розпакованої теки wxWidgets.

Примітка. На момент написання роботи описаний вище алгоритм для ОС Windows 7 на самому останньому кроці (і для наочного, і для програмного способу налаштування) завершувалося повідомленням про помилку з кодом 5702. На жаль, цю проблему не вдалося подолати, навіть дотримуючись наявних у глобальній мережі рекомендацій щодо встановлення оновлень ОС Windows. І про такі невдалі спроби інших користувачів також є повідомлення в Інтернеті з натяками на некоректність самих оновлень.

У разі відсутності у читача можливості працювати на ПК під керуванням ОС Windows 10 радимо використати ОС Ubuntu Linux або LinuxMint.

У подальшому описі використано рекомендації для Ubuntu, перевірені на LinuxMint 18.1 Serena редакції Mate. Цю редакцію вибрано з міркувань невибагливості графічного середовища Gnome2, що дуже важливо для бюджетних організацій, у тому числі загальноосвітніх закладів освіти. У разі використання версії операційної системи Linux, яку не підтримують у поточний момент, процес встановлення CodeBlocks 16.01 (пункт 1 подальшого опису) не буде успішно завершено внаслідок спроби звертання до джерел, відсутніх у відсутніх репозиторіях.

Алгоритм встановлення програмного забезпечення для ОС Ubuntu/Mint Linux

1. Встановити CodeBlocks – бажано останньої версії, використавши PPA — *Personal Packages Archive* — персональний архів пакунків.

На відміну від репозиторіїв Ubuntu, де зібрано велику кількість паунків, PPA містить лише окрему версію певної програми. Для встановлення програм з PPA використовують не менеджер програм, а *Термінал*. Для версії 16.01 потрібно виконати по черзі такі вказівки:

```
sudoadd-apt-repository\
ppa:damiem-moore/codeblocks-stable
sudoapt-getupdate
sudoapt-getinstallcodeblocks\
codeblocks-contrib
```

2. Завантажити зі сторінки wxwidgets.org/downloads/ архів (SourceCode) останньої або останньої стійкої версії — файл `wxWidgets*.tar.bz2`. Тут і далі зірочкою * позначено версію. Наприклад, 3.1.1.

3. У середовищі Терміналу перейти у теку із за-

вантаженим архівом і розпакувати архів вказівкою Терміналу:

```
cd Завантаження
tar -xvfwxWidgets*.tar.bz2
```

4. Зайти у теку — результат розпакування архіву — вказівкою Терміналу:

```
cd wxWidgets*
```

5. Запустити конфігуратор для перевірки наявності всіх залежностей вказівкою Терміналу:

```
./configure
```

Примітка. При встановленні ПЗ, поширюваного архівами tar.bz2, при відсутності деяких файлів буде виведено відповідне повідомлення. У цьому випадку перед виконанням вказівки ./configure рекомендують використати такі вказівки Терміналу:

```
або ./bootstrap
або ./autogen.sh
або
aclocal
autoheader
automake --gnu --add-missing --copy --foreign
autoconf -f -Wall
```

6. Запустити процес компіляції (збирання) програми вказівкою Терміналу:

```
make
і дочекатися припинення виведення повідомлень, серед яких не має бути повідомлення про помилку (error).
```

7. Завершити встановлення, використавши такі вказівки Терміналу:

```
sudo apt-get install checkinstall
sudo checkinstall
```

і погодившись на всі запити, тобто ввівши "у" або натиснувши клавіші Enter.

8. Скопіювати приклади (понад 80, є опис англійською на сторінці docs.wxwidgets.org/3.1/page_samples.html), розташовані у теці samples всередині теки wxWidgets* — результату розпакування архіву сирцевого коду wxWidgets*.tar.bz2, використавши такі вказівки Терміналу

```
cd samples
make
```

9. Пересвідчитися у дієздатності програм-прикладів (хоча б деяких), запустивши на виконання виконувані файли, створені всередині тек access, animate, ... , xti, розташованих всередині теки samples. Наприклад, calendar має виводити обрані дати (крім першої, вибраною самою програмою) — див. рис. 2–4.

10. Пересвідчитися у можливості компілювання програм-прикладів окремо, виконавши всередині відповідної теки таку вказівку Терміналу:

```
g++ -g назва_коду.cpp $(wx-config --cxxflags --libs) -o/назва_виконуваного_файлу
```

Наприклад, для теки calendar ця вказівка може мати такий вигляд:

```
g++ -g calendar.cpp $(wx-config --
```

```
cxxflags --libs) -o tobedone
```

Примітка. Назва теки збігається з назвою коду — назвою файлу програми-прикладу, який містить ця тека. Переміщення теки, що містить програму-приклад, з wxWidgets*/samples, може призвести до помилки компіляції. Станом на червень 2018 року не було конфігураційних файлів проектів CodeBlocks для програм-прикладів.

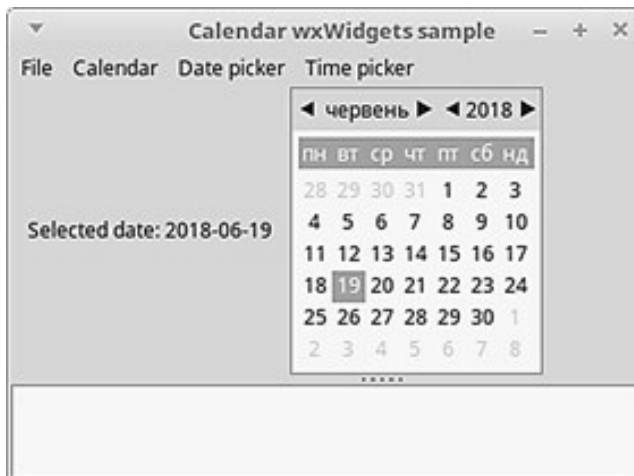


Рис. 2. Вікно програми calendar одразу після виклику 19 червня 2018 року

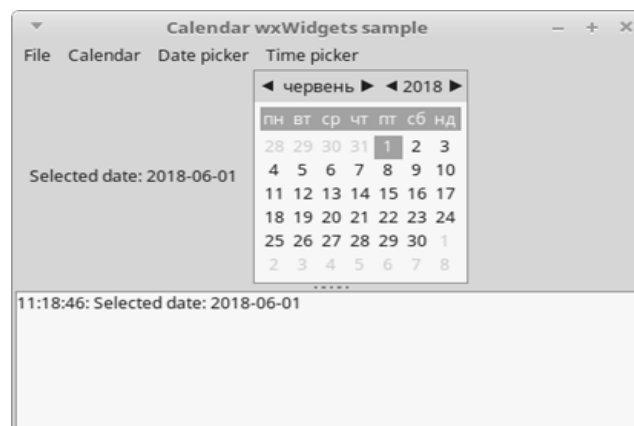


Рис. 3. Вікно програми calendar після першого вибору дати 1 червня 2018 року

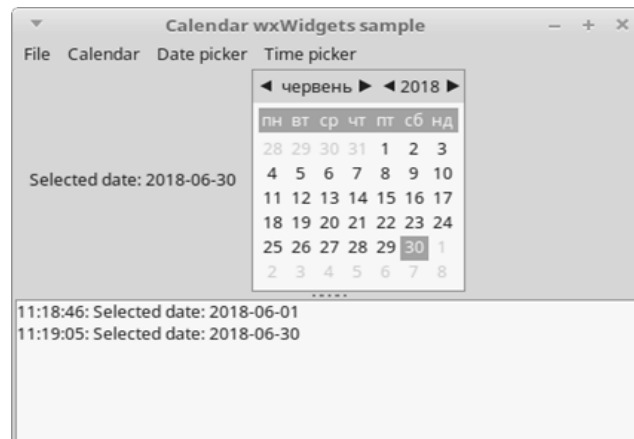


Рис. 4. Вікно програми calendar після другого вибору дати 30 червня 2018 року

Ознайомлення з кодами програм-прикладів потрібно вважати лише першим кроком знайомства з можливостями WxWidget.

Наступний крок може бути таким:

- створити нову теку у директорії wxWidgets*/samples;
- скопіювати у створену теку код cpp програми-прикладу;
- модифікувати у середовищі довільного текстового редактора код cpp у з метою досягнення схожого, але іншого результату;
- компілювати код, використовуючи у створеній теці вказівку *Терміналу*:

```
g++ -g назва_коду.cpp $(wx-config --sxxflags --libs) -o \
назва_виконуваного_файлу
```

Для того, щоб модифікувати програму-приклад у середовищі CodeBlocks, потрібно виконати певні дії.

Алгоритм завантаження програм-прикладів у середовище CodeBlocks з метою подальшого редагування має такий вигляд [3]:

1. Скопіювати теку з прикладом у зручне для роботи місце.

2. У середовищі CodeBlocks створити порожній проект з такою самою назвою, що має скопійована тека:

- використати вказівку меню *File / New / Project...*;
- вибрати *wxWidgets project* і натиснути кнопку *Go*;
- вибрати *wxWidgets 3.0x* і натиснути кнопку *Next*;
- ввести назву проекту у полі *Project title*, вказати розташування скопійованої теки у полі *Folder to create project* і натиснути кнопку *Next*;

• при бажанні ввести автора, адреси його електронної пошти та сайту і натиснути кнопку *Next*;

вибрати:

Preferred GUI — Builder None;

Application Type — Frame Based;

і натиснути кнопку *Next*;

• обрати *Create "Release" configuration* і натиснути кнопку *Next*;

• серед *Other Options* обрати *Create Empty project* і натиснути кнопку *Finish*.

3. Використати вказівку меню *Project / Add files...*, вказати усі наявні у теці файли з розширенням cpp і h (для описаного вище прикладу лише calendar.cpp), і натиснути кнопку Відкрити.

4. Підтвердити вибір кроку 2, натиснувши кнопку *Гаразд*.

5. Скопіювати файл *sample.xpm* з теки wxWidgets*/samples у теку створеного проекту.

Примітка. Такий файл, але для іншого зображення, можна створити і самостійно у середовищі редактора растрової графіки. У цьому випадку, можливо, доведеться (у середовищі текстового редактора) виправити його другий рядок, щоб він набув такого вигляду:

```
static const char *const sample_xpm[] = {
```

6. У середовищі CodeBlocks відкрити всі приєдна-

ні файли, знайти рядок:

```
#include "../sample.xpm"
```

і видалити з нього дві крапки, тобто надати йому такого вигляду:

```
#include "sample.xpm"
```

(для файла calendar.cpp це 60-ий рядок). Інакше кажучи, врахувати розташування файла sample.xpm не безпосередньо зовні теки проекту, як надано архівом wxWidgets, а в теці проекту.

7. Наступні дії виконати лише при роботі з ОС Windows:

- клацнути правою кнопкою миші на проекті;
- в контекстному меню вибрати *Build option*;
- перейти на вкладення *Linker settings*;
- натиснути кнопку *Edit* і у вікні діалогу *Edit library* натиснути кнопку з трикрапкою і вказати розташування бібліотеки *libwxmsw31u.a*, після чого натиснути кнопку *Гаразд*;
- у вікні діалогу *Project build options* натиснути кнопку *Гаразд*.

8. При бажанні змінити у рядках коду (подано для calendar.cpp):

```
// Create the main application window
MyFrame *frame = new MyFrame(
wxT("Calendar wxWidgets sample"),
wxPoint(50, 50),
wxSize(450, 340));
```

заголовок вікна wxT, верхній лівий кут вікна wxPoint і розміри вікна wxSize.

9. Скомпілювати програму і запустити її на виконання.

Література

1. Ильин Е. Компиляция библиотеки wxWidgets в Code::Blocks. — Режим доступа: <https://jenyay.net/Programming/Wx>

2. Моон С. Як скомпілювати і встановити wxWidgets при Ubuntu/Debian/LinuxMint. — Режим доступа: <https://www.binarytides.com/install-wxwidgets-ubuntu/>

3. Уотсон Г. Приклади wxWidgets. — Режим доступа: <http://web.eng.fiu.edu/watsonh/eel3160/ProgOfWeek/PW6.pdf>

4. Лапінський В. В. Проблема вибору першої мови програмування — сьогоднішнє бачення // Комп'ютер у школі та сім'ї. — 2014. — №1 (113). — С. 14–17

References. Translation and transliteration

1. Ilyin E. Compiling the wxWidgets library in Code::Blocks. — Access code: <https://jenyay.net/Programming/Wx>

2. Moon S. How to compile and install wxWidgets on Ubuntu/Debian/Linux Mint. — Access code: <https://www.binarytides.com/install-wxwidgets-ubuntu/>

3. Watson H. wxWidgets Samples. — Access code: <http://web.eng.fiu.edu/watsonh/eel3160/ProgOfWeek/PW6.pdf>

4. Lapinsky V. V. The problem of choosing the first programming language - the present vision V. V. Lapinsky. // Computer in school and family. - 2014 - # 1 (113). - P. 14-17

НАСТРОЙКА ВЗАИМОДЕЙСТВИЯ WXWIDGETS И CODEBLOCKS ДЛЯ ОСОВРЕМЕНЕНИЯ ОБУЧЕНИЯ ИНФОРМАТИКЕ

Рудик Олександр Борисович

*кандидат фізико-математичних наук, доцент,
доцент Інституту післядипломної педагогічної освіти
Київського університету імені Бориса Грінченка,
rudykob@gmail.com*

Анотація. . Стаття посвящена аналізу и решению проблем, связанных с изменением парадигмы изучения информатики в школе: существенным ростом количества учебных часов на изучение программирования. Обоснован выбор программного обеспечения для учебного процесса, представлены конкретные рекомендации по его установке и описаны возможные проблемы такой установки.

Ключевые слова: событийное и объектно ориентированное программирование, C++, CodeBlocks, wxWidgets

SETTING UP WXWIDGETS AND CODEBLOCKS INTERACTIONS FOR PROVIDING THE STUDYING OF INFORMATICS TRAINING

Alexander Rudyk

*Ph.D (physics and mathematics), associate professor of the
Institute of Postgraduate Pedagogical Education
of Boris Grinchenko Kyiv University,
rudykob@gmail.com*

Annotation. The article is devoted to the analysis and solving of problems related to the change of the paradigm of studying computer science at school: a significant increase in the number of study hours for studying programming. The choice of software for educational process is substantiated, concrete recommendations for its installation are given and the possible problems of such installation are described. .

Keywords: event oriented and object-oriented programming, C++, CodeBlocks, wxWidgets.

Продовження у наступних номерах

* * *

УДК 37.014.63:37.014.55

АНАЛІЗ БАЗОВИХ ЗАСАД УПРАВЛІННЯ ОПОРНИМ ЗАКЛАДОМ ОСВІТНЬОГО ОКРУГУ

Калініна Людмила Миколаївна

*доктор педагогічних наук, професор, завідувачка відділу економіки
та управління загальною середньою освітою Інституту педагогіки НАПН України
lmykkalinina@gmail.com
ORCID ID 0000-0003-0534-6089*

Топузов Михайло Олегович

*кандидат економічних наук, старший науковий співробітник відділу економіки
та управління загальною середньою освітою Інституту педагогіки НАПН України.
ORCID ID 0000-0001-7890-5959*

Анотація. У статті розглянуто результати аналізу досвіду організації та діяльності різноманітних закладів освіти. Показано, що якість освіти та її вартість у розрахунку на одного суб'єкта навчання залежить від кількості учнів у навчальному закладі. Зазначені залежності є нелінійними, їх екстремуми, а саме мінімум вартості навчання і максимум якості освіти не збігаються, але їх зближення вказують на можливість оптимізації процесу. Результатом зазначеного може бути суттєвий і позитивний суспільний ефект – покращання якості освіти. Сформульовано деякі поради щодо інформатизації освітнього процесу та управління ним.

Ключові слова: управління, заклад освіти, освітній округ, електронні освітні ресурси .

Постановка проблеми. Підготовка до виконання певного завдання, пов'язаного з модернізацією суспільних інститутів, завжди передбачає:

а) попередній аналіз необхідних властивостей мак-

симально можливої кількості складників системи, яку передбачається створити, виокремлення їх суттєвих властивостей, можливо – побудову графу зв'язків, аналіз можливих шляхів на цьому графі як відображення