

РОЗРОБКА МАКРОСІВ КОМП'ЮТЕРНИХ ТЕСТІВ

Терещенко Ю.П.

Високий рівень розвитку інформаційних технологій в освіті дозволяє активно використовувати їх для організації та проведення контролю знань учнів у різних формах навчання. Комп'ютерне тестування дозволяє за найкоротші терміни перевірити знання великої групи учнів; виявити прогалини у вивченні конкретного навчального матеріалу й використовувати отримані результати для управління ходом навчального процесу; застосувати методи математичної статистики для оцінки ступеня засвоєння навчального матеріалу; отримати об'єктивну оцінку знань учнів у різних навчальних закладах.

У даній статті розглядається процес створення тестів у середовищі Power Point на основі необхідного набору макросів. Маючи шаблон такої презентації, тест може створити будь-який учитель, що вміє працювати з програмою PowerPoint.

Перш ніж розпочати роботу над створенням тесту, потрібно скласти невеликий сценарій того, як повинна працювати така презентація. Складемо наш тест таким чином, що за кожну правильну відповідь учень отримає один бал. На початку презентації учневі буде запропоновано ввести своє ім'я. Це потрібно буде для того, щоб записати отриманий результат у файл. Кожен слайд, окрім запитання, міститиме варіанти відповіді на нього. Варіантами можуть бути текст, малюнки, формули, будь-який об'єкт, який можна імпортувати в презентацію. Коли учень, натиснувши ліву клавішу миші, вибере правильний варіант, з'явиться повідомлення про підтвердження відповіді. У кінці тестування учень зможе переглянути оцінку.

Тест розроблений у середовищі MS Microsoft PowerPoint 2003, але створений шаблон працює й у середовищі MS PowerPoint 2007.

Перш за все нам потрібно створити презентацію, у якій, окрім слайдів із запитаннями, будуть присутні ще три слайди: слайд із назвою тесту, з якого розпочинатиметься тестування, слайд із переглядом оцінки та заключний слайд тесту. Якщо тест складатиметься із дванадцяти запитань, то для нашого тесту потрібно буде 15 слайдів. Тестовий шаблон буде побудовано так, що на етапі його створення в учителя не буде ніяких заборон щодо оформлення. Єдиною вимогою буде збереження першого та двох останніх слайдів, зазвичай, з можливістю їх оформлення.

Створивши відповідну кількість слайдів, оформивши їх візуально, переходимо до програмування. До першого слайду додаємо командну кнопку, з допомогою якої буде відкриватись форма для введення імені учня. Приклад першого слайду показано на рис. 1.

До передостаннього та останнього слайдів теж додамо командні кнопки, які будуть слугувати відповідно визначенням кількості оцінок та завершенням те-

сткування. Приклади слайдів можна побачити на рис. 2 та 3 відповідно.

Для того щоб додати командну кнопку, потрібно в меню **Показ слайдів \ Керуючі кнопки** вибрати потрібну кнопку та клацнути лівою клавішею миші по слайду.

Перш ніж ми розпочнемо наповнювати слайди із завданнями, слід потурбуватись про макроси, які забезпечать нам автоматизацію перевірки. Для роботи з макросами нам потрібно відкрити Редактор Visual Basic. Це можна зробити за допомогою команди **Сервіс \ Макрос \ Редактор Visual Basic** або ж натиснувши комбінацію клавіш **Alt+F11**.

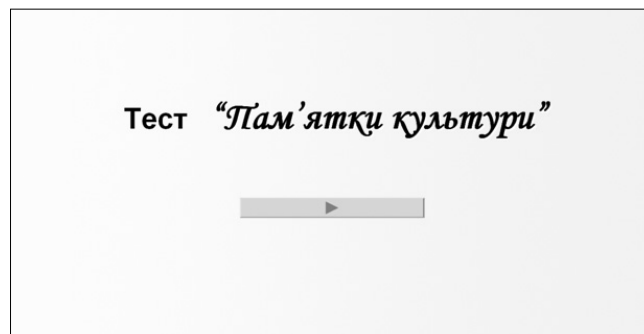


Рис. 1. Перший слайд тесту

Якщо Редактор VBA не встановлений на вашому комп'ютері, необхідно його встановити. Для цього виконайте команду **Пуск \ Налаштування \ Панель управління \ Установка и удаление программ**. У списку встановлених програм виділіть Microsoft Office та натисніть кнопку **Изменить**. У діалоговому вікні **Установка Microsoft Office** виберіть **Добавить** або **Удалить компоненты** та натисніть кнопку **Далее**. У наступному діалоговому вікні поставте галочку біля надпису **Расширенная настройка приложений**, натисніть кнопку **Далее**. У наступному вікні натисніть значок «+» біля надпису **Общие средства Office**. Знайдіть у відкритому списку надпис Visual Basic для при-

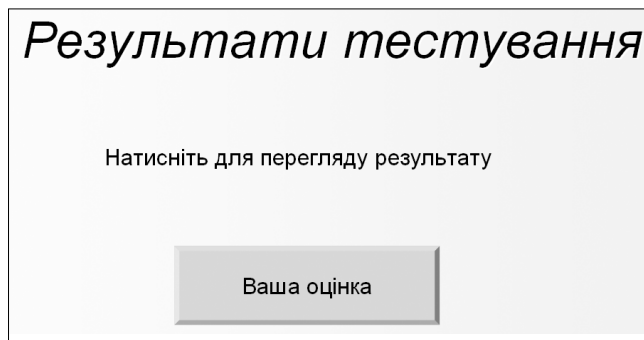


Рис. 2. Слайд із кнопкою для відображення оцінки

Тестування завершено

Вихід

Рис. 3. Останній слайд презентації

ложений, натисніть позначку диска біля надпису та у випадаючому списку виберіть **Запускати з мого комп'ютера**. Натисніть кнопку **Обновити** для завершення установки. Завантажте знову Excel та перевірте, чи встановився у вас Редактор Visual Basic.

Тепер познайомимось із редактором Visual Basic детальніше, головне вікно якого зображено на рис. 4.

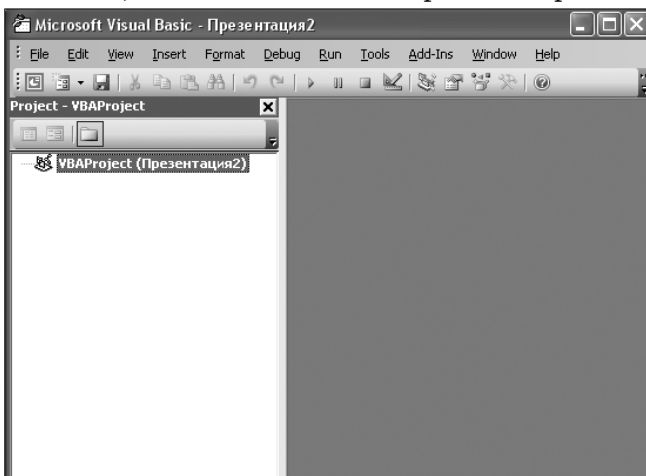


Рис. 4. Редактор VBA

За замовчуванням вікно редактора містить активне вікно менеджера проекту (Project — VBAProject) та поле, у якому відображається вікно модуля з макросами. Щоб створити модуль, необхідно в меню **Insert** вибрати команду **Module** (рис 5).

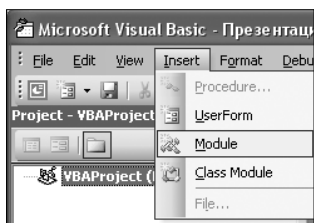


Рис. 5. Створення модуля

Після цього у вікні менеджера проекту з'явиться модуль (рис. 6) з ім'ям **Module1**. Тепер у правій частині вікна редактора Visual Basic можна створювати процедури (макроси), які потрібні для роботи тесту.

Перший макрос, який нам потрібен, макрос для введення імені учня, що тестується.

Для цього ми скористаємось функцією **InputBox**. Щоб запам'ятати введене прізвище, скористаємось змінною рядкового типу **NameU**. Змінну оголосимо в коді модуля так, щоб вона була доступна для інших процедур. Тому в коді модуля запишемо рядок **Public NameU As String**. Також для нашого тесту потрібні інші змінні: кількість правильних та неправильних відповідей та змінна для визначення номера запитання, на яке повинен відповідати учень.

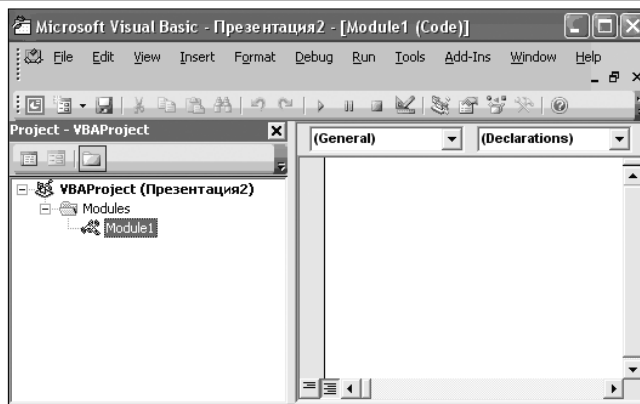


Рис. 6. Створення нового модуля

Public NameU As String — змінна для імені учня.

Public KolN As Integer — змінна для кількості неправильних відповідей.

Public KolP As Integer — змінна для кількості правильних відповідей.

Public list As Integer — змінна для визначення номера поточного запитання.

Створимо процедуру для ідентифікації учнів.

Лістинг 1

'Ідентифікація учня

```
Sub form_Open()
Do While NameU = «»
NameU = InputBox(«Ідентифікація», «Введіть своє ім'я», «»)
Loop
list = list + 2
SlideShowWindows(Index:=1).View.Next
End Sub
```

У даній процедурі команда введення імені учня знаходиться в середині циклу. Це зроблено для того, щоб учень обов'язково увів своє ім'я.

Тепер займемось підрахунком правильних відповідей. Для цього в модуль вставляємо процедуру **bal()**.

Лістинг 2

'Процедура для підрахунку правильних відповідей

```
Sub bal()
Dim prompt, button, title, helpfile, context, result
prompt = «Ви впевнені?»
button = vbYesNo + vbQuestion
title = «Підтвердження відповіді»
helpfile = «Demo.hlp»
context = 20
If list = SlideShowWindows(1).View.Slide.SlideIndex Then
result = MsgBox(prompt, button, title, helpfile, context)
If result = vbYes Then
' Якщо натиснута кнопка Yes
KolP = KolP + 1 ' Замість цифри 1 підставити
кількість балів
SlideShowWindows(Index:=1).View.Next
list = list + 1
End If
If result = vbNo Then
' Якщо натиснули No. В даному випадку учень
може повторити вибір варіанту
End If
Else
If list = 0 Then
SlideShowWindows(1).View.GotoSlide (1)
Else
SlideShowWindows(1).View.GotoSlide (list)
End If
```

```
End If
End Sub
```

У даній процедурі передбачено можливість того, що учень почне тестування не з початку, а з довільного слайду, тому ведеться перевірка поточного номера слайду та перехід до наступного.

Процедура для підрахунку неправильних відповідей.

Лістинг 3

‘Підрахунок неправильних відповідей

```
Sub schetN()
Dim prompt, button, title, helpfile, context, result
prompt = «Ви впевнені?»
button = vbYesNo + vbQuestion
title = «Підтвердження відповіді»
helpfile = «Demo.hlp»
context = 20
If list = SlideShowWindows(1).View.Slide.SlideIndex Then
result = MsgBox(prompt, button, title, helpfile, context)
If result = vbYes Then
KolN = KolN + 1
SlideShowWindows(Index:=1).View.Next
list = list + 1
End If
If result = vbNo Then
End If
Else
If list = 0 Then
SlideShowWindows(1).View.GotoSlide (1)
Else
SlideShowWindows(1).View.GotoSlide (list)
End If
End If
End Sub
```

Для відображення оцінки запишемо таку процедуру.

Лістинг 4

‘Демонстрація оцінки

```
Sub ozenka()
If list = SlideShowWindows(1).View.Slide.SlideIndex Then
MsgBox «Ваша оцінка» & KolP * 12 \ 8 & « » & «Неправильних відповідей: » & KolN
Rezultat ‘Виклик процедури для запису результату у текстовий файл
SlideShowWindows(Index:=1).View.Next
list = list + 1
Else
If list = 0 Then
SlideShowWindows(1).View.GotoSlide (1)
Else
SlideShowWindows(1).View.GotoSlide (list)
End If
End If
End Sub
```

У даній процедурі відбувається виклик процедури Rezultat, яка записує результати проходження тесту в текстовий файл. Її лістинг подано нижче.

Лістинг 5

```
Sub rezultat()
Dim frFile ‘ задаємо довільний файл
frFile = FreeFile ‘ визначаємо довільний файл
t = «c:\rezult.txt» ‘!!!!!!(Вказати повний шлях до файлу)
Open t For Append As frFile ‘ Відкриваємо для дописування результатів
```

```
‘ записуємо прізвище учня із змінної NameU та оцінку зі змінної KolP
Print #frFile, NameU & « » & «оцінка» & STR(KolP)
Print #frFile, «-----»
Close frFile
End Sub
```

Увівши всі процедури в модуль, залишається зберегти зміни та перейти до створення слайдів із запитаннями. Оформлення запитання та варіантів відповідей довільні. Для того щоб визначити правильний варіант відповіді, необхідно виділити текст чи малюнок, натиснути праву кнопку миші та в контекстному меню вибрати команду **Настройка действия**. У вікні, що відкривається, необхідно вибрати команду **Запуск макроса** та у випадіючому списку вибрати макрос bal (рис. 7).

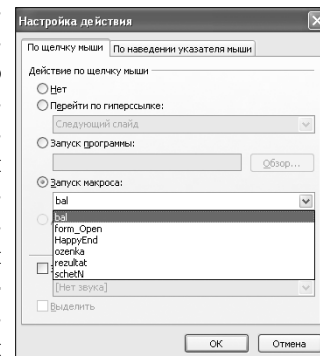


Рис. 7. Вибір макросу

Для неправильного варіанту необхідно вибрати макрос schetN. Після створення завдань не забудьте перевірити формулу для визначення оцінки. Формула записана в процедурі ozenka() у вигляді KolP* 12\8, де число 12 є максимальним балом, а число 8 кількість запитань. Змінюючи відповідну кількість балів та запитань, можна визначати оцінку за тест. Останнім штрихом є визначення макросів для керуючих кнопок із першого та двох останніх слайдів. Для першого слайду необхідно задати макрос form_Open для кнопки з оцінкою ozenka, для останнього слайду вибрати макрос HarryEnd.

Після виконання всіх операцій збережіть презентацію у вигляді шаблону. Тепер її можна використовувати для створення інших тестів.

Для того щоб PowerPoint працював із макросами, необхідно увімкнути середній або низький рівень безпеки макросів.

Останній штрих — збереження вже готового тесту у вигляді демонстрації. У цьому випадку учні відразу зможуть приступити до тестування.

Пакет програм Microsoft Office є дуже потужним інструментом. У запропонованій статті розглянуті далеко не всі можливості, які можна використати під час створення тестових чи навчальних програм. Сподіваюсь, що запропонований матеріал принесе користь учителям та їхнім учням і буде стимулом до отримання нових знань.

Література

1. Биллинг В.А. Средства разработки VBA-программиста. Офисное программирование. Том 1 / Биллинг Владимир Арнольдович. — М.: Издательско-торговый дом «Русская Редакция», 2001. — 480 с.
2. Использование Visual Basic 6 / [Боб Ресельман, Ричард Писли, Вайн Пручняк, Эрик Смит]. — Пер. с англ. — К.: М.; СПб. Издательский дом «Вильямс», 1999 — 608 с.: ил., ISBN 5-8275-0035-6 (рус.)
3. За матеріалами сайту <http://www.vbnet.ru/>.