UNIVERSITY OF Hull

Teaching Programming using SNAPS

Rob Miles

Department of Computer Science

Introduction

- About me
- The Problem of Teaching Programming
- Encouraging Engagement
- Snaps overview
- Snaps demo

Welcome to my world..

- Lecturer in Computer Science in University of Hull
- Learned to program using punched cards and Algol 60
- Been teaching programming for over 20 years
- Currently teach C# programming to our First Year students
  - Very keen on Python too
  - ..and the Arduino..
  - ..and the Raspberry Pi
  - Microsoft MVP for Windows Platform Development
- Blogs at www.robmiles.com

We teach C#

- We've taught C# as a first language for a while
  - It's heavily based on Java
  - Same syntax as C, C++ and Java
  - Strongly typed
  - Flexible
  - Good for future employment
- Later in our course we teach C++, Prolog, assembler etc

The Problem of Teaching Programming

- It is impossible to teach programming
  - Just like it is impossible to teach someone how to ride a bicycle
- All you can do is tell them the fundamental principles and then get them to have a go
  - And support them when they fall off
- Students can get a bit upset when they discover that we can't "tell them the correct answer to the question so that they can learn it, write it down and get good marks"

Starting to Learn to Program

- To get students to learn how to program you have to get them to practice problem solving
- This is especially important in the early stages when they are learning how to create and sequence the program statements they use to construct their solutions
- Until they are confident that they can do this, there is no point in progressing any further with the teaching
- Being able to derive a solution and express it as a sequence of individual statements is the fundamental skill that underpins everything else in programming

Context is Key

- The most important aspect of learning how to solve problems with software is the context in which the problems to be solved are framed
- If the students don't understand what their program is trying to achieve, they will never be able to construct a workable solution to the problem
- The problem context must be familiar, relevant and amenable to a software solution
- It also helps if it is novel/fun ☺

Establishing Context

- If you are teaching computing for use by a Physicist you can place the problem in this context:

  "Write a program to perform Ohm's Law calculations."

- In a Computer Science course you can use line of business problem solving

- Either way it is important to instil professional principles from the start:

  "Your program should behave sensibly if the user enters a resistance of zero."

Instilling Professionalism from the start

- Ensuring that a program is "correct" from the start is a very context for programming concepts:
  - Input validation is a great way to start thinking about conditional statements
  - Repeating requests is a great way to start thinking about asking for data if the input is invalid
- This also gets the student thinking that code to "keep the solution sane" should be part of any program from the start
- From this we can move into the necessity for test (and what test actually means)

After context, motivation

- The next most important thing is motivation
- Students want to learn things that they can do to impress people, especially other students
  - Writing things on the screen using `Console.WriteLine` doesn't usually achieve this…
- Things that have been found to impress include:
  - Writing games
  - Writing programs that make Minecraft worlds
  - Turning lights on and off (as per Arduino)
- How about making programs you could sell?

Application writing

- Writing "proper" applications is difficult for beginners
- To write an application the student must be au-fait with some advanced programming concepts
  - Objects: the display system will be object based
  - Events: input will be provided on the back of events
  - Tasks: many systems use asynchronous operation
  - API: each interface will require objects to be created and marshalled
- By the time the student can understand these they will already be good programmers
  - Or they will have given up before this stage.....

Enter "Snaps

Simple .NET Application Program Services

```
string guestName;
guestName = Snaps.ReadString("What is your name");
Snaps.SpeakString(guestName);
```

- Snaps is a library of "snap together" library components in C# that are easy to understand and use
- They are packaged and used in exactly the same as "`Console.Readline`" but they provide easy access to interesting resources
- And they are entirely synchronous in use

The Importance of Synchronicity

- It is fundamentally important that the Snap methods all run synchronously
- This is so that we can teach the fundamentals of programming using them
  - This requires some interesting programming behind the scenes….
- We will need to introduce events and asynchronous operation later in a programming course
  - We can do some of this by showing how the Snaps actually work

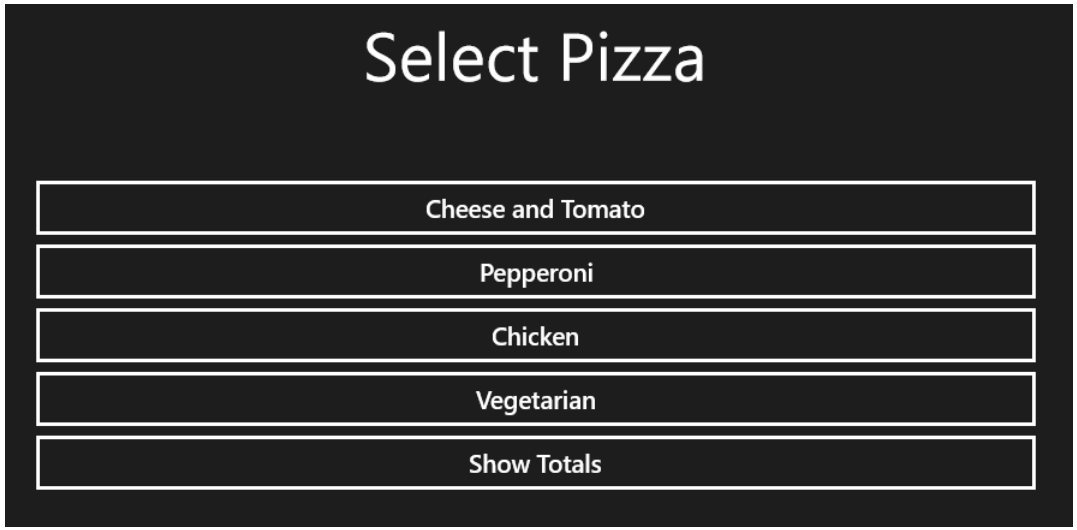
Snap Demo: Weather Forecaster

```
string forecast = Snaps.GetWeatherForecastDescription("UKXX0476", 1);
Snaps.SpeakString("The forecast is " + forecast + ".");
int forecastTemp = Snaps.GetForecastTemperatureInFahrenheit("UKXX0476", 1);
Snaps.SpeakString("The temperature will be " + forecastTemp.ToString());
```

- A snaps program can get the weather forecast for a location (temperature and conditions) and then speak the result
- Warn you to wear a coat or sunscreen
- Plenty of scope for customisation
  - Display different weather pictures etc (Snaps apps can display embedded or web based graphics)

Snap Demo: Pizza Picker

```
string toppingChoice = Snaps.SelectFrom5Buttons("Cheese and Tomato",
"Pepperoni",  "Chicken", "Vegetarian", "Show Totals");
if (toppingChoice == "Cheese and Tomato")
    cheeseAndTomatoCount = cheeseAndTomatoCount + 1;
```



- The Snaps library provides a synchronous way of creating and reading user input that makes menu driven code easy to create

Snaps Deployment

- Snaps has been written as a set of libraries for Windows 8.1 desktop and phone
  - Programs written using Snaps can be deployed for both platforms
  - It is planned to provide versions for Android and iOS
- It will be deployed in the form of a Visual Studio project template
- The Snaps framework will be released as Open Source on GitHub

Snaps Rules

- Rules for making Snaps:
  1. Easy to use (no reference parameters, throw or return status, use exceptions when appropriate)
  2. Atomic (one method per function, no communication via side effects)
  3. Synchronous (method only returns to the caller when the task it was called to perform has been completed).
  4. Portable (works on all platforms in exactly the same way)

Snaps and Teaching

- We will be using Snaps in our teaching starting 2015-2016
  - A Microsoft Press book is being written to underpin the course
- The idea is to get students working with the Snaps and generating (and maybe even publishing) applications using the library – and maybe even writing their own snaps
  - Embedded Snaps
  - Snaps that use the Marvel comic API
- Snaps provides a strong context for generating solutions and learning sequential programming

Resources

- For Snaps newsletter:

  http://www.robmiles.com/snaps

- Complete C# course:

  http://www.csharpcourse.com

- Python course:

  http://www.wrestlingwithpython.com

- Arduino labs:

  http://www.robmiles.com/arduino

- World Famous Blog:

  http://www.robmiles.com