Teaching programming
the PRIMM way

11th and 14th February

I Love Computing Conference

Sue Sentance, University of Cambridge and Raspberry Pi Foundation
Overall aims of this session

- If you’ve never heard of PRIMM, then I’m hoping this session will give you an introduction!
- If you’re a PRIMM expert, I’ll hopefully leave plenty of time for you to share your experiences so we can all learn together!
- There are a couple of activities so have a pen and paper ready!
Starter task: You’ll need a pen and paper!
An activity for you - you will need pen and paper!

Look at this Python (turtle) code

[Discuss with your partner]

What happens when the code runs?

Draw the output

We can’t do this part today!
from turtle import *

def square():
    for counter in range(4):
        forward(100)
        right(90)

square()
left(45)
square()

Now run the code
Did you get the same?
If not, what is different about your answer?

We’ve just demonstrated the P and R of PRIMM
Working in pairs

If you had been able to work in pairs I might ask you:

● What discussions did you have?
● Did you help each other?
● What did you learn?
How did PRIMM come about?

What we noticed in class (in 2016)

Students were spending a whole lesson copying in a program (block-based or text) provided by the teacher

OR

Students were given a problem to solve and had to start from a blank screen

Students had their hands up for a long time, while the teacher had to circulate and fix individual syntax errors

Students were working alone

Could this be changed to .... ?

Starting from a program that existed in the world

Delaying responsibility / ownership until students could understand what they were working on

Spending time on concepts and how programs actually worked

Understanding a program at different levels of abstraction

Talking about programs with each other and using out-loud explanations

Students went off task
Key principles of PRIMM

Predict-Run-Investigate-Modify-Make

Communicate and collaborate
Read and understand code
Gradually take ownership
The PRIMM approach

**Predict** – given a working program, talk about it in pairs or groups. What do you think it will do?

**Run** – run it and test your prediction

**Investigate** – get into the nitty gritty. What does each line of code mean? Lots of activities to try here: trace, annotate, explain, talk about, identify parts, etc.

**Modify** – edit the program to make it do different things

**Make** – design a new program that uses the same nitty gritty but that solves a new problem
Predict

Big Ed is your friendly chatbot.

Run

Investigate

Modify

Make

With a partner, spend time reading the code below. PREDICT exactly what you think will happen.

Adapted from Teach Computing Curriculum, Year 7, Programming Unit Part 1, Lesson 3
PRIMM Walkthrough

Predict

Now run the code.
- Were your predictions correct?

Run

- Did anything surprise you about what happened when the code was executed?

Investigate

- Did you miss anything out? If so, what?

Modify

Make

The programs are shared with students, they don't have to be re-created.

ncce.io/biged2
Answer the following questions with your partner by experimenting with the code in order to investigate how it works.

How do the following two blocks of code relate to each other?

Temporarily remove the following block:

What has this changed about the program when you run it?
Why do you think this is?

Below define ask_name, there are two variables being used. What are their names?
Now place the `ask What’s your monster name? and wait` block back into the code, but place it after the `set name to answer` block:

Why do you think it only says “Hello” and not “Hello “ and the name you entered?

What can you learn from this?

Ask different types of questions.
PRIMM Walkthrough

All of the code needed to make Big Ed ask about your favorite music has been included in the program. Place the blocks below into the appropriate places in the subroutine.
Create your own character to have a conversation with.

The character can be anything you want. Maybe related to a favourite hobby or sport?

The character should:
- introduce themselves
- ask the user 2 questions
- respond appropriately based on the answers the user gives.
Focusing in on Investigate

What are good questions to ask?

What questions might you ask learners to help you gauge their understanding?

What questions could you ask your students which would prepare them to write something similar?
### We can frame questions around the Block Model

(this is another talk!)

<table>
<thead>
<tr>
<th></th>
<th>(M) Macro structure</th>
<th>(R) Relationships</th>
<th>(B) Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Understanding</strong></td>
<td>Understanding the overall structure of</td>
<td>Understanding the sequence of</td>
<td>Understanding the function of</td>
</tr>
<tr>
<td><strong>the</strong></td>
<td>the program text</td>
<td>function calls, object</td>
<td>the block of code.</td>
</tr>
<tr>
<td><strong>goal/purpose</strong></td>
<td>of the program</td>
<td>sequence diagrams</td>
<td></td>
</tr>
<tr>
<td><strong>program</strong></td>
<td>in the current context</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>in the</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>current</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>(M) Macro structure</th>
<th>(R) Relationships</th>
<th>(B) Blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Language</strong></td>
<td>Language elements</td>
<td>Operation of a block</td>
<td>Understanding the function of</td>
</tr>
<tr>
<td><strong>elements</strong></td>
<td></td>
<td>or function.</td>
<td>the block of code.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>(T) Text Surface</th>
<th>(P) Program Execution</th>
<th>(F) Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Architecture/Structure</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Relevance/Intention</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
We can frame questions around the Block Model

<table>
<thead>
<tr>
<th>Block</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Macro structure</strong></td>
<td>Annotate the whole program showing where functions are called and defined.</td>
</tr>
<tr>
<td><strong>Relationships</strong></td>
<td>Identify the scope of a variable</td>
</tr>
<tr>
<td><strong>Blocks</strong></td>
<td>Draw a block around an if statement</td>
</tr>
<tr>
<td><strong>Atoms</strong></td>
<td>List all the integer variables</td>
</tr>
<tr>
<td><strong>Answer</strong></td>
<td>(T) Text Surface</td>
</tr>
</tbody>
</table>

The Block Model

| Architecture/Structure | Relevance/Intention |
Activity - Asking questions

What "Investigate" questions could you ask students to see if they understood the code - and to prepare your students to write something similar?
Does PRIMM make a difference?
Our research

Pilot study 2017 - iteration of materials, proof of concept

Mixed-methods study 2018 - evidence that PRIMM impacted on learning outcomes

Follow-up study 2020 - PRIMM and classroom talk
PRIMM provides support for students

“[previously] the lower ability just couldn't access it at all. And they would just sit there absolutely confused and flummoxed. And you’d spend your whole time going over and over and over. Whereas the PRIMM scaffolds it all in, so there's never a giant leap. Even the Make task [...] is one step further than what they've already done. So, there's a sense of security.” (secondary teacher)
Students work together

“Particularly at the beginning of the PRIMM lesson or series of lessons, there’s very much a sense of discovering together what it does and how it works and so an awful lot more peer conversation going on.”
Developing the ability to ask good questions

“I could almost hear myself in their voices ... I would hear - But why does that work? Why is yours better than mine? How can I make mine look like yours and still make it work? Yours looks more efficient, explain to me why it’s more efficient.” (secondary teacher)
Students can find a language to express themselves...

"They really get stumped with the difference between a parameter and an argument. And understanding what a parameter is and a return value. I think that’s where you’d see the most difficult thing that they talk about. Which is why we try and give them a language because the language helps them to express themselves better when they’re talking about it. And also it helps them I think to have a mental model of what that is, if you give it a name."

(secondary teacher)
Students get engaged in productive discussion...

The children are generally talking to each other and often working through some questions together, deciding what they want to say as their answer. Then there’s a lot of discussion around how to get things to work [primary teacher]

Particularly at the beginning of the PRIMM lesson or series of lessons, there’s very much a sense of discovering together what it does and how it works and so an awful lot more peer conversation going on. [primary teacher]
As a teacher, I can model programming terms...

I’m very strict on vocabulary on class so I tend to model quite a lot from the front and model the correct vocabulary ... I think PRIMM helps that when you’re doing "what do we think this does" and then you can go and talk about it, insisting that they talk in the correct vocabulary. And then just letting them have those conversations with each other. (secondary teacher)
Students talk less ... and at a more advanced level ...

"I find that I’m having less time talking to the whole group, because we can get the programs running in the first place, so I’m talking to the group probably less ... but ... I’m talking at a more advanced level to the whole group, but for less time. When I’m asking questions, they’re usually much more useful and probing questions ... " [secondary teacher]
Resources using PRIMM ...

- primmportal.com
- Programming Pedagogy in Secondary Schools rpf.io/secondarypedagogy
- Hello World - Big Book of Pedagogy helloworld.cc
- Programming Essentials in Scratch rpf.io/pathfinders-scratch
- Teach Computing Curriculum KS4 rpf.io/tcc-ks4

Other teachers’ resources

- Phil Bagge’s PRIMM resources
- Andy Colley’s PRIMM unit of work
- Jon Bridgman’s PRIMM lessons
- Liane O’Kane Christmas PRIMM lesson
- Pam Jones’ PRIMM page
Questions?

Let us know how you get on!

Sue Sentance sue@raspberrypi.org @suesentance

https://www.surveymonkey.co.uk/r/UKICTS

An aside: I’m (when I have time) writing a short book(let) about PRIMM. If you’re interested in giving feedback on a draft let me know in the chat.