Design in Primary Computing Programming

- Algorithmic design, Appearance design, Structural Design
- Overlap
- Proficiency
- Primary Domain Competency
- Planning Focus
- Should every project have appearance design?
- Moving Forwards

Algorithmic design, Appearance design, Structural Design

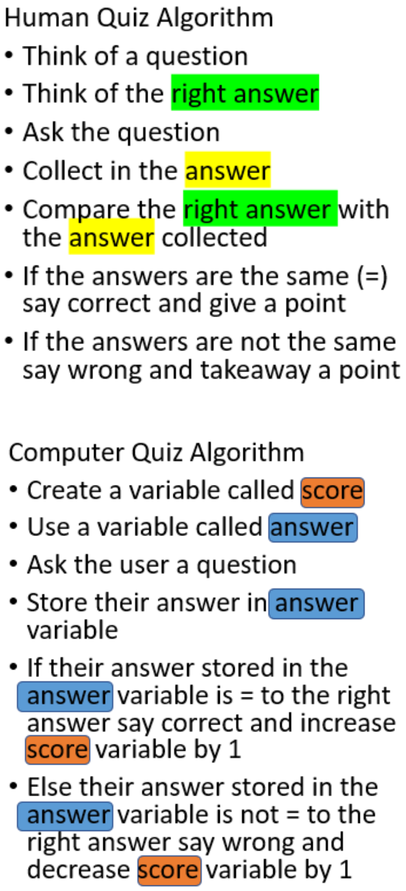
Computing programming design has many elements.

Algorithmic design can be described as thinking through the primary functions of the project. The steps or rules to achieve the primary function that can be turned into code. At primary level this can include basic data structures such as variables or simple lists which we might separate as *data structure design* at a later point.

Appearance Design can be thought of as all the aspects of the project that relate to how a project will look and how the user will experience it. User interface design is a big part of this. It might also incorporate screen and physical art work and primary teachers may wish to incorporate aspects of the art curriculum into appearance design.

Structural Design includes any structural design and technology that we need to include in a project and If we are programming physical objects (like buzzers and motors) then we would include electronic wiring planning as part of structural design.

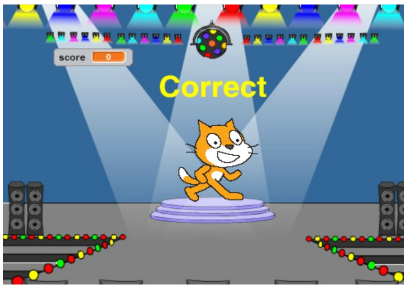
Typically in a simple quiz, the job of making a quiz question producing the right answer and keeping a score belongs to the algorithmic design process. What type of questions the programmer chooses to test, how the quiz is introduced, the user experiences and the chosen effects once the user has got



Human Quiz Algorithm
- Think of a question
- Think of the right answer
- Ask the question
- Collect in the answer
- Compare the right answer with the answer collected
- If the answers are the same (=) say correct and give a point
- If the answers are not the same say wrong and takeaway a point

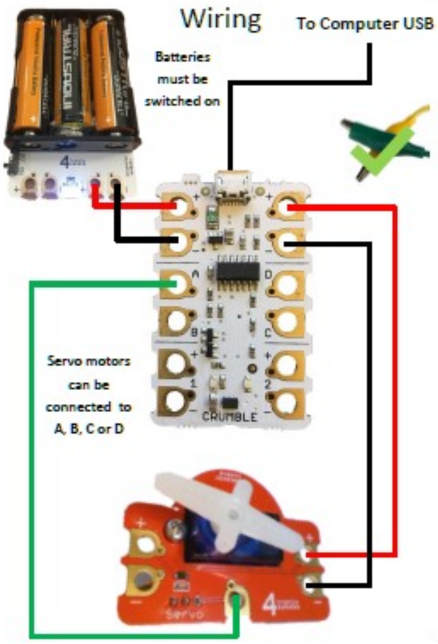
Computer Quiz Algorithm
- Create a variable called score
- Use a variable called answer
- Ask the user a question
- Store their answer in answer variable
- If their answer stored in the answer variable is = to the right answer say correct and increase score variable by 1
- Else their answer stored in the answer variable is not = to the right answer say wrong and decrease score variable by 1

Algorithmic design in a Scratch Quiz



Appearance design in a Scratch Quiz

Appearance design elements in an animated animal project


Wiring
To Computer USB
Batteries must be switched on
Servo motors can be connected to A, B, C or D
CRUMBLE

Structural design, wiring a Servo Motor

the answer right or wrong belong more generally to appearance design although they might also necessitate elements of algorithmic design.

When creating an animated animal using the Crumble programming language and servo motors it is easy to delineate between all three design types. Appearance design involves what choice of animal and what art work we will use to make our creations interesting. Algorithm design and subsequent programming involve thinking through what we want the servo motor to do and how we want it to interact with a button or distance sensor. Structural design involves how to wire these in such a way that every component works together What materials we will use to mount the servo and art work on and how these will be attached.

Overlap

There is always going to be overlap between all three design areas. Some appearance design choices will necessitate algorithmic design changes. Some structural design choices will limit or extend appearance and algorithmic design and visa versa.

Proficiency

A pupil may be more proficient in one area whilst lacking in skills in one or both other areas.

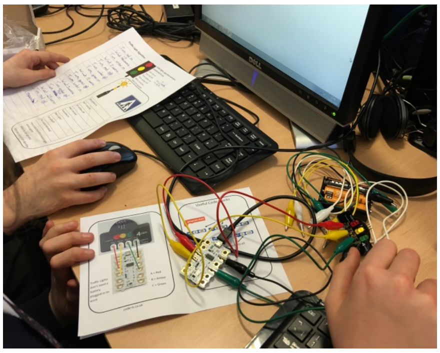
Primary Domain Competency

Primary domain competency for algorithmic design rests within the computing science domain. Appearance design shares domains with a host of creative disciplines such as art, graphics design etc depending on the materials and techniques advocated. The primary domain competency of structural design is design and technology.

Planning Focus

As in all learning situations, the teacher may choose to allow greater choice and independence in one aspect and provide greater guidance or support in others. A rounded curriculum would hope to draw on programming projects from a wide variety of genres and give pupils a chance to develop all three aspects of design planning. Currently, due to a focus on computational thinking it is fair to say that appearance design has taken a back seat in screen based programming projects.

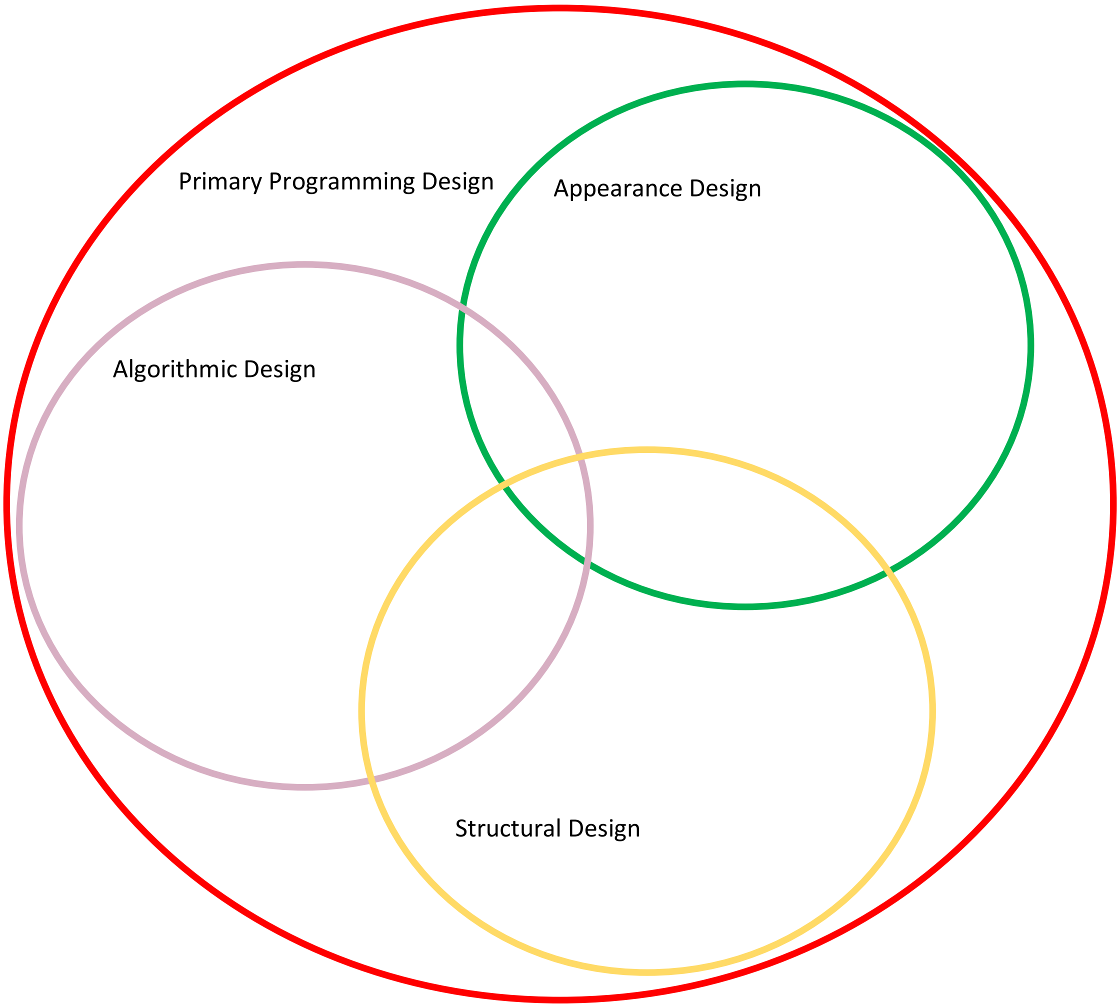
Should every project have appearance design?

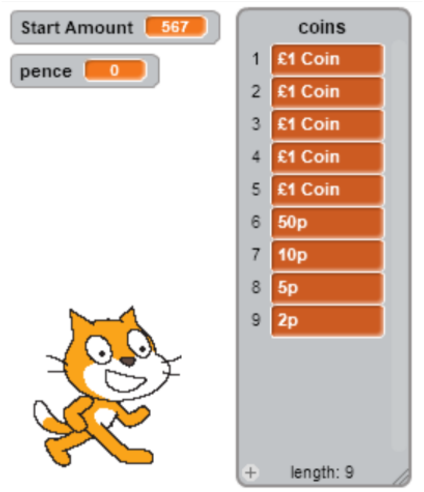


In year four or five I do a short Crumble project which involves thinking through the traffic lights sequence in detail before turning this precise algorithm into code on our crumble microcontrollers and traffic light accessories. It would be possible to mount the traffic light, hide the wiring and concentrate on the appearance design but to be honest in this project it never feels like it needs this, algorithmic design and the wiring part of structural design are enough.

However the animated animal, mentioned in the previous section would be significantly poorer without the appearance design elements.

My Year six pupils are currently finishing their independent Scratch platform games. Appearance design and algorithmic





Start Amount 567

pence 0

coins

1	£1 Coin
2	£1 Coin
3	£1 Coin
4	£1 Coin
5	£1 Coin
6	50p
7	10p
8	5p
9	2p

length: 9

A change machine programmed in Scratch

design are equally important to such a visual programming outcome.

When programming a change machine with Year 5 this year the beauty and elegance of it was wrapped up in the elegance of the algorithm and appearance design is a much smaller part of the project.

A similar complex project of designing and programming a clock needed a much bigger appearance design aspect once we moved from digital clock to analogue. Clock face hands benefit from good appearance design thinking.

I would summarise by saying that if the project has a significant visual aspect then time spent on appearance design is time well spent. If the beauty of the project is primarily in the algorithmic complexity then appearance design can take a back seat, for that project. If a project needs structural design then time spent developing this can help develop pupils engineering thinking.

Moving forward

I am going to make the appearance design element of projects much clearer in rewrites or new planning where I think it would benefit from this.

Thanks

My thanks to Jane Waite for her article in Hello World about design, a useful phone and email conversation and her recent article which triggered this re-think. All emphasis and errors are mine.

You can read her article here.

http://www.ijcses.org/index.php/ijcses/article/view/23

Phil Bagge 8th April 2018