# Computer Science Project Proposal

List at least 15 success criteria for your project: *(continue overleaf)*

Users need to think strategically in order to pass the game

Game can run smoothly and provides good user experience

Program does not contain serious bugs

OO is used

It contains some complex algorithms

Code is as elegant as possible

Describe your project proposal:

For my project, I am going to make an educational game using Python and its extension library Pygame. The game will try to illustrate some advanced computing algorithms that may or may not be taught in A-level computer science. The idea is that some computing problems all have a very interesting story behind lies the complicated algorithms, so I would like to illustrate these algorithms vividly for users and hopefully this could encourage more people to learn these algorithms that can seem quite daunting at first.

Name: Vincent Liu

Proposed Language: *(you will need permission first if not Python)* Python

Describe the stakeholders for your project:

Despite every effort is made to ensure this game is as interesting as possible, this is game is not designed for absolute beginners who want to learn more about computing algorithms. The target audience would be those who have basic coding skills and, ideally, have completed an A-level course or equivalent and want to study further in this area.

Describe the influences and inspiration for your project. Talk about the research you have done and how this has influenced the features for your own project.

As mentioned above, this is an educational game aimed to encourage more people who have already obtained some level of coding skills and computational thinking and want to study further and learn more advanced algorithms. This is idea was inspired by the UK Bebras Challenge, which allows the user to interact with the computer to learn some computational ideas.

What resources will you use to design and write your project?

PyCharm IDE

Online resources of Pygame tutorials

Success criteria continued:

The game is as interactive as possible

This game can encourage people to learn more about these algorithms rather than deter them

Solutions are straightforward for players to follow

Stories are interesting to most players

Principles behind games are logically strict

**Implementation**

Your project is 20% of your final grade. In order to give yourself the best chance of the highest mark, it is critical you return in September with a roughly working alpha version of your game. Creating your game should be an enjoyable exercise and the more you get done during the vacation, the easier life will be next term.

Many of you have indicated you would like to see your predicted grade rise for UCAS applications. The completion of the alpha version is a necessary condition, although not the only one, of your grade rising. If you return in September having done nothing, **there is no chance whatsoever** of your predicted grade being increased.

Use this page and the next to think about how you would design some of the algorithms you have mentioned in your success criteria above. Number each one and write the pseudo-code, with comments, for each below.

**Implementation (continued)**

**Screen Design**

What form will the user interface take? What will be visible to the user (e.g. high score, map, HUD, instructions, etc). Draw a design of the screens you will have in your game below and on the next page. Annotate each screen with an explanation.

**Screen Design (continued)**

**Feedback** is vital to the development of your project. Present your ideas to the class and your teacher. Record below the comments that were made and how they have influenced the project development. What changes or improvements will you now be making in light of these comments?

Use this page for further notes on any of the sections above