

Project

- Write a C++ programme that reads a set of shapes for a given file.
- The shapes are defined in the file by a name string, a colour string, followed by a set of x,y points that describe the shape or some other data depending on the shape. All coordinate should be positive!
- There are two files to test your programme with: “data1.dat” and “data2.dat”. “data2.dat” contains errors in some lines that your programme should cope with. Start with “data1.dat”
- Colour is restricted to the following: red, green, blue, yellow, white, black. Your classes should store this in an appropriate manner.
- Add a function `void print_name()` to print the name of the shape
- Add a function that calculates and returns the width of the object. To do this you need loop over the vectors to find the minimum and maximum x-point.
- Add a function that calculates and return the height of the object. To do this you need loop over the vectors to find the minimum and maximum x-point.
- Add a function that calculates the area (as define by the bounding box, i.e., use the width and height and calculate the area from this.)
- Print all the shape names, widths, heights and areas
- Sort the order of the shapes into area

Data formats

Shape name	data
Point	int x, int y
Line	colour, Point, Point
Polyline	colour, An unknown number of Points
Rectangle	colour, Point , Point, Point, Point
Circle	colour, Point centre, double radius

Suggested Solution

1. See Lecture 5 for the Points/Shape example. This could form the base of you solution.

2. Create a class to hold `Points` and a class enum to hold the colour.
3. Create a base class called `Shape` and a set of classes to hold the shapes, that subclasses `Shape`
4. `void print_name()` should be a pure virtual member of `Shape`
5. Test these classes, describe them in the report (why did you choose the design, what's so good about it) and show how you tested your code.
6. Add the width and height functions. Test them. Describe them in the report and show how you tested them.
7. Process the file `data1.dat`. Read each line at a time. Get the name of the shape, and then store the list of `Points` in a vector. Test it and
8. Implement the sort method.
9. Process the file `data2.dat` - this file has errors so you will need to cope with them. I would suggest ignoring lines with errors. This bit is hard.

Marking

Marks are given for:

- The use of an appropriate class design.
- Correct use of abstract classes
- Correct types for storing colour information.
- All classes should have appropriate data and function access modifier levels and appropriate “getter” and “setter” functions.
- Error detection and handling correctly - (you could for example subclass `std::Exception` to create your own exception class, use `std::runtime_error` or use another appropriate method. File processing uses flags rather than exceptions. See lecture notes.)
- All resources must be cleaned up correctly before the programme exists.
- All classes and functions should have the correct comments included.
- The correct use of RAII principles.
- The correct use of keywords (`const`, `virtual`, `override` etc)

What to hand in

- A report describing your code. Please explain each section of your code. Briefly outline your thinking behind why you chose the methods you did. Explain what any function

does that you wrote or that you use. You can either do this in the report or as comments in your actual code. There is a word limit of 3500 words for the report. This is overkill and I do not think you will need this many. Most of the marks are for the code!!

- Record your progress in your report. As you develop a routine, test it by for example printing values, and cut a paste these into your report. You should say what values you are expecting to be printed and why.
- This will be an electronic submission. Your report should be in **pdf only**. No word files please. This should be contained in a zip file that includes all of your code.
- If you have a working version that partially completes the task and an uncompleted version I would suggest submitting both in different directories in your zip file. Please describe this in your report.

Marks breakdown

- Class design - 55%
- Main function (storing of shapes, etc) - 15%
- Report - 5%
- Reading data1.dat 15%
- Reading data2.dat 10%