Coursework 1: hand in by Friday, 19th March through Moodle

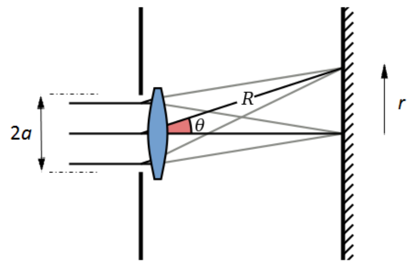
- These exercises should help to improve your programming ability as well as assess it.
- Your coursework submission should take the form of a **word-processed report**. It should include **printouts of your C programs** (e.g. clear screenshots are sufficient) and their **results**, and **discussion** of the approach/methods used to **design** and **test** your programs. Write also which C-compiler you used (e.g. cc compiler from Linux-Bath machine).
- **60% of the marks** will be given for **sound, working programs**.
- **Additional 40% of marks** will be given for **clarity and good programming style**, for more sophisticated programs and for the **presentation, discussion** and **very good report writing style.**
- So always make sure that you get a good, straightforward program working first, before you try to get it to do something more complicated. Any program that partly carries out the given task will be worth marks, even if it does not fully do the job.
- Electronic copies of the **report (in PDF format!) and the C-program files** should be submitted through Moodle.
- **Note.** Your report and your programs must be your own work. University regulations require that all your submitted work will be tested with anti-plagiarism software and anyone suspected of plagiarism or collusion will undergo the University's formal disciplinary process.

1. Diffraction Limit of a Telescope (35 marks)

When light from a star (essentially a point-source), with wavelength λ, passes through the circular aperture of a telescope, the image produced at the focal plane consists of a circular diffraction pattern … a central bright dot, surrounded by a series of concentric rings of diminishing brightness.

The intensity of the light is given by $I(r) = I_0 \left(\frac{2\,J_1(x)}{x}\right)^2$,

where $x = ka\sin(\theta) = \frac{2\pi}{\lambda}\, a\, \frac{r}{R}$ and $J_1(x)$ is a Bessel function.



The Bessel functions, $J_m(x)$ are given by

$$J_m(x) = \frac{1}{\pi}\int_0^{\pi} \cos(m\theta - x\sin\theta)\, d\theta$$

where m is a non-negative integer and $x \geq 0$.

(a) Write a C function `J(m,x)` that calculates the value of $J_m(x)$ using the trapezium rule, with N = 10000. Use your function in a program to calculate the Bessel functions $J_0(x), J_1(x)$ and $J_2(x)$ as a function of x from $x = 0$ to $x = 20$. Make a plot of your results.

(b) Write a second program that calculates the diffraction pattern (using the equation given above) of a circular lens, for which the focal ratio (i.e. $R/2a$) is 10. Calculate the pattern for the direction defined by the parameter *r*, as shown in the figure above. Plot the diffraction pattern for values of *r* covering the range ± 25 µm. Pick a sensible wavelength for the calculation.

The only math library functions you can use are `pow()`, `sin()` and `cos()`.

2. Root-Finding (35 marks)

The finding of roots of equations via numerical methods is necessary where analytic solutions do not exist (an example from quantum physics would be the energy of a particle in a finite one-dimensional quantum well). The aim of this exercise is to numerically locate the roots of **one** of the following equations. The equation that you need to solve numerically in this exercise depends on the **last digit** of your five-digit candidate number, e.g. 26789 → 9.

0, 1 or 2: $f(x) = x^3 + 5x^2 + x + \sin(x)$

3, 4 or 5: $f(x) = 5\sin^3(x) + (x+1)^2 + \frac{1}{(x+1)^2}$

6 or 7: $f(x) = \frac{1}{(x+1)^2} + \frac{1}{(x-1)^2} - 10$

8 or 9: $f(x) = \cos^2(x) + x^3 - x^2 - 0.291064$

(a) Discuss the limits of numerical precision and possible sources of rounding errors in root finding.

(b) Design (using pseudocode or a flowchart) a program to find the real roots of the equation. The user should specify the required precision (number of significant figures) of the final numerical result. Your program should take account of the issues you discussed in part (a). If your algorithm requires use of the derivative of the function, you must use a numerically calculated value of the derivative at each point (i.e. not an analytical expression for the derivative).

(c) Translate your program design of question 2(b) into a real C program, debug and run it. Give a listing of the program and the values of any roots that you obtain (with comments on any difficulties encountered and solved).
Discuss the testing of your program in order to demonstrate that it produces answers to the correct level of precision.

Note: You are not required to give analytic solutions to the roots of the given equation.

3. Multivariable Equations (30 marks)

Design and encode a C program to solve the **one** of the following sets of simultaneous equations. Again, the set of simultaneous equations that you need to solve in this exercise depends on the **last digit** of your five-digit candidate number, e.g. 26789 → 9.

0 or 3: $4x - 2x^2 + 2y^3 = 1$ and $4y^4 + x^4 + 4y = 4$

1 or 4: $4xy^2 + 2xy = 2$ and $4x^2y + x^2 = 2y$

2 or 5: $x^2 + xy = 10$ and $y + 3xy^2 = 57$

6 or 8: $(x+1)^2 + (y+1)^2 = 25$ and $xy + y^2 = 5$

7 or 9: $x^2 + 1 = y$ and $y = 3\cos(x)$

There are several ways you might do this. In Lecture 4, we discussed the mathematics behind solving such equations in vector form. With only two equations, a simple first step would just be to write out the separate equations, then solve them using the techniques developed in Q2.

Again, one should use a numerical approach to calculate the derivatives, and the user should be able to specify the required precision (number of significant figures) of the final numerical result.