MEDICINE AND MEASUREMENT

INTRODUCTION

Pandy's test is an old test (still used by vets) which is performed on cerebrospinal fluid (CSF) to detect elevated levels of proteins, particularly large water-soluble proteins called globulins. The test involves adding a drop of CSF to a small amount of Pandy's solution; if the test is positive, the resulting liquid becomes cloudy or opaque. This cloudiness of a liquid is called turbidity.

According to Wikipedia, a positive Pandy's test may indicate one or more of the following pathological conditions:

- Diabetes mellitus
- Brain tumour (Meningioma, Acoustic neuroma or, Ependymoma)
- Encapsulated brain abscess
- Spinal cord tumour
- Multiple sclerosis
- Acute purulent Meningitis
- Granulomatous Meningitis
- Carcinomatous Meningitis
- Syphilis (protein may be normal if longstanding)
- Guillain–Barré syndrome [Infectious polyneuritis] (Protein rises after 5–7 days)
- Cushing's disease
- Connective tissue disease
- Uremia
- Myxedema
- Cerebral haemorrhage

Today's task is to build a simple turbidity meter that works by measuring how much light is absorbed by a solution. The basic idea is as follows:

In our case, the light will be provided by an LED whose brightness can be controlled, the sample is actually a dilute solution of white emulsion paint, and the detector is a light sensor on a little computer called an Engduino that we're about to learn to program.
WHY CONFUSE COMPITING AND SCIENCE – THEY’RE DIFFERENT, SURELY

Well, actually, a very large amount of the science and medicine done today is done using computers. There are computers in just about all modern measuring devices and computers are essential in processing all sorts of medical and scientific data, from MRI scans to collisions in the large hadron collider, the behaviour of sheep and the ways that proteins fold. Put simply, learning to be a good modern scientist involves learning to program.

So, with no further ado...

INTRODUCING THE ENGDUNIO

The Engduino is just a simple, small, computer that is capable of measuring quite a lot of interesting things about the real world and that has a bunch of LEDs that can be used to provide feedback to the user.

The Engduino board comes pre-fitted with some sensors and some LEDs. There are many more sensors than we need today, but importantly, there is a light sensor on the front.

For your first exercise you are going to write a really simple program and then upload it to your Engduino board. A programming window should be open for you already and should look a bit like the picture below. If not, just ask.
Now we’re going to write a program.

Type the following into the white box.

```c
#include <EngduinoLEDs.h>

void setup()
{
  EngduinoLEDs.begin();
}

void loop()
{
  EngduinoLEDs.setAll(GREEN);
  delay(1000);
  EngduinoLEDs.setAll(OFF);
  delay(1000);
}
```

Notice that All is spelt with an upper case A and lower case l.

Now let’s make it run.... make sure your Engduino is plugged in¹ and switched on.

Click on the upload button:

![Upload button](image)

This checks your code for some types of error, turns it into something that a machine can understand, and then uploads it to the Engduino board. If there are errors they will appear in the black section in red.

Voila – your first Engduino program. Can you make it flash twice as fast? In what units do you think the number in the delay(…) statement might be?

¹ If you are not in a lab class, make sure that in Tools > Board > LilyPad Arduino USB is selected, and that under Tools > Serial Port the serial port that’s connected to your board is selected.
USING THE LIGHT SENSOR

The light sensor measures light flux – it returns a higher value when more light lands on the sensor. Let’s change the program we wrote before to do the following:

```c
#include <EngduinoLight.h>

void setup()
{
    EngduinoLight.begin();
}

void loop()
{
    int v = EngduinoLight.lightLevel();
    Serial.print("Light value: ");
    Serial.println(v);
}
```

Press the upload button again. After it has uploaded, click on the button in the top right hand corner of the screen that looks like:

![Serial Monitor icon]

It should say ‘Serial Monitor’ next to it when you move the mouse pointer over it.

You will see values printed on the screen (which range from 0 to 1023). Cover up the light sensor – what happens? Ask the demonstrators to plug in the LED – what happens to the value now?

Change the program slightly so that it takes one reading per second.

DOING THE EXPERIMENT

There are two sets of vials – one slightly smaller than the other. The one you use will be determined by the size of the hole on your board. The set labelled A-E are larger vials, and those labelled a-e are smaller vials.

CALIBRATION

Each sensor is slightly different and two different sensors will often give slightly different readings in the same light conditions. What’s more, in this equipment, the LEDs will all be slightly different brightnesses, and the boards will be in different parts of the room with different amounts of room light or daylight falling on them.

So first we need to calibrate our equipment. To do that, we use samples with known values of turbidity and measure the light values for those. Sample A is pure water and is defined to have a turbidity of 0 Turbidity Units (TU). Sample E is defined to be opaque and to have a value of 10 TU.
We are now going to take some readings and find out how turbid our samples are.

(i) Take (at least) five light readings for each vial and record these carefully (shake the vial if a precipitate has settled). Calculate the average of those readings. This is the value you should use on the graph. The order you do this in is not important, so if you’re sharing vials, use ones that are not currently being used by others.

(ii) Take the light readings for samples A and E and put these points on the graph. Draw a straight line between them. This is your calibrated line.

(iii) Take the light readings for vials B-D and, using your graph, figure out how turbid the unknown vials are and record this value.

<table>
<thead>
<tr>
<th>Vial</th>
<th>Reading 1</th>
<th>Reading 2</th>
<th>Reading 3</th>
<th>Reading 4</th>
<th>Reading 5</th>
<th>Average</th>
<th>Turbidity</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>10</td>
</tr>
</tbody>
</table>

QUESTIONS...

A. Why do we take five readings rather than just one?

B. Does twisting the vials so they face slightly different ways affect the reading? If so, why might that be and what should we do about it?

C. What about daylight or the light from the room lights? The light sensor is sensitive to all forms of light, regardless of how it arrives – how could we make a better instrument?

D. Compare your results with a group who used a different sized vial. Are the values you got for turbidity different? Actually, the samples in B and b, C and c, D and d, are the same turbidity, so what have we forgotten? What could we standardise to fix the problem?