

Project Baseline

J T Davenport

Teaching Fellow
Aston University
Birmingham, United Kingdom
t.davenport@aston.ac.uk

M Peters

Director Engineering and Science Foundation Programmes
Aston University
Birmingham, United Kingdom
m.peters@aston.ac.uk

G Aggarwal

Mathematics Tutor
Aston University
Birmingham, United Kingdom
g.aggarwal@aston.ac.uk

Conference Key Areas:

- Open and Online Engineering Education
- Curriculum Development
- Mathematics and Engineering Education

Keywords:

- Online mathematics education

Project Baseline

INTRODUCTION

The School of Engineering at Aston University offers a Foundation Year for students whose qualifications are different from the normal UK university applicants. The main groups are:

- UK applicants who have not followed the GCSE – A' level route
- UK applicants who have not achieved satisfactory grades at A' level
- International applicants who have studied in a different educational system

The cohort arrives with a wide range of skills in mathematics and quite wide expectations in terms of what it means to learn at university. Some understand that it is up to them to learn but they are a minority. The majority are not independent learners and expect to be lead through the syllabus in the way they were when they were at school.

Many students arrive with an inadequate grasp of basic arithmetic and algebra.

An additional factor is the inability of most applicants to use mathematics in any way other than the way they have been taught.

Our task:

1. Take students from diverse starting points and get them to a minimum end point ready to begin a degree course in Engineering
2. Create independent learners
3. To create mathematical confidence based on students' ability to:
 - a. Take a problem
 - b. Abstract the key features
 - c. Make defensible assumptions
 - d. Produce a result
 - e. Sanity check^[1] the result

1. HISTORY

A programme of baseline tests was started in 2012 in an attempt to identify which students needed additional support. Students were given a test in the first week of term. A detailed analysis was created for each student from the results of the test giving students specific information about areas that needed improvement.

It took a lot of time to create individual learning analyses for the cohort of 70 and not a single student took advantage of the additional resources made available to them. The consequences were far too often seen in the final results. The test results provide a good resource for the teaching team both in terms of understanding the requirements of individual students and to help creating groups.

The Foundation Year has seen significant growth in student numbers. The number of students has more than tripled since 2012.

The teaching team wanted to continue using baseline tests but the increasing student numbers created an unmanageable workload for the available staff. The team decided to develop a computer application to manage the baseline tests and, with a flash of inspiration, they called Baseline.

Baseline started as a Windows application running on the university network but as student numbers grew larger than the number of work stations in the main computer laboratories it was decided to move the code to a web application so students can access it in their own time and on their own devices.

The web version of Baseline is designed to:

- Provide learning resources
- Provide sample tests with appropriate feedback
- Manage tests
- Track the engagement and attainment of individual students
- Report on the progress of the cohort and of individual students

2. SETTING THE SYLLABUS

Like many universities, Aston has a drop-in centre for students that provides help with subjects like essay writing, technical English and mathematics. Consultation with mathematics staff in the drop-in centre identified the syllabus given below.

Experience in the drop in centre has shown that students arrive with a patchy knowledge of mathematics. They are often able to carry out complicated algorithms like integration by parts but when faced with expressions like $10 - 6 + 8$ will give the answer -4.

- Basic arithmetic including:
 - Number lines,
 - Number placement
 - Addition, subtraction, multiplication and division
 - Order of precedence
 - Products of prime numbers
 - Common sets of numbers
- Algebra including:
 - Terms and like terms
 - Brackets
 - Equations
 - Factorising
- Fractions, decimals and percentages
 - Addition, subtraction, multiplication and division of fractions
 - Addition, subtraction, multiplication and division of decimals
 - Addition, subtraction, multiplication and division of percentages
 - Converting between fractions, decimals and percentages

3. SOFTWARE ARCHITECTURE

Baseline runs on a Linux server using PHP for the server side code, MySQL for the database and JavaScript, HTML and CSS for the client side application. Slim Framework is used for server side security. Bootstrap and MathJax are used for client side formatting. The landing page is shown in *Fig. 1*.

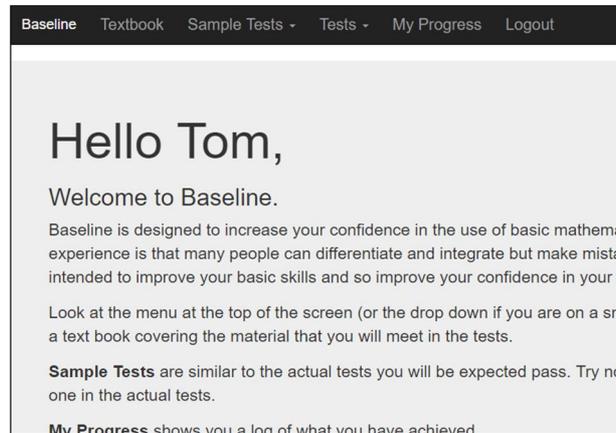


Fig. 1. Baseline landing page

3.1 Textbook

Advice from Aston's drop-in centre was to create the content of the Baseline Textbook starting at a basic level. Topics like number lines, number placement, algorithms for addition and subtraction are covered. The intention was to provide a complete, non-patronising resource to students that would allow them to learn or re-learn topics they should have grasped in primary or early secondary school but had somehow missed.

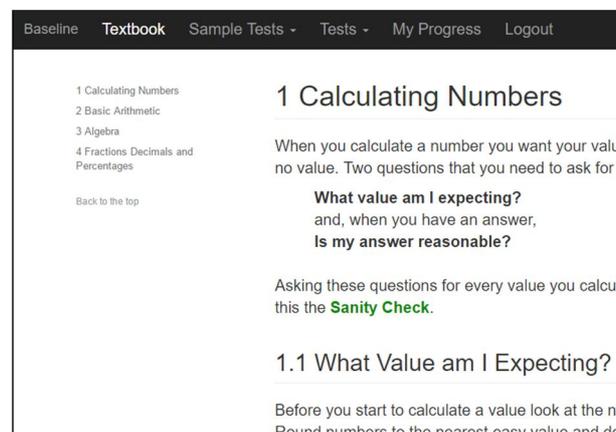
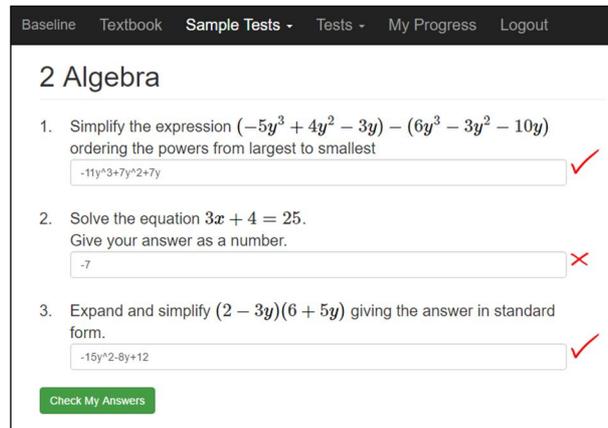


Fig. 2. Baseline textbook

It is expected that students would know most or all of the content but Baseline is a resource where they could review and revise the fundamental principles.

3.2 Sample Tests

The sample test questions and answers are stored in a format called JSON^[2] in the Baseline database. JSON (JavaScript object notation) is human-readable and more concise than XML. Questions are mixture of text and LaTeX^[3]. The text font is set by the .css^[4] files and MathJax^[5] re-renders the LaTeX. See the Fig. 3.



The screenshot shows a web interface for a sample test. At the top, there is a navigation bar with links: Baseline, Textbook, Sample Tests (selected), Tests, My Progress, and Logout. Below the navigation bar, the page title is '2 Algebra'. There are three questions listed:

1. Simplify the expression $(-5y^3 + 4y^2 - 3y) - (6y^3 - 3y^2 - 10y)$ ordering the powers from largest to smallest. The input field contains $-11y^3+7y^2+7y$ and has a red checkmark to its right.
2. Solve the equation $3x + 4 = 25$. Give your answer as a number. The input field contains -7 and has a red X to its right.
3. Expand and simplify $(2 - 3y)(6 + 5y)$ giving the answer in standard form. The input field contains $-15y^2-8y+12$ and has a red checkmark to its right.

At the bottom of the questions, there is a green button labeled 'Check My Answers'.

Fig. 3. Sample test questions provided by Baseline

3.3 Tests

When a test is selected the content is generated on the server so no two tests are the same. The code for generating question 2 on the algebra paper is given in Fig. 4.

```
// =====  
function getALQ02($inNum) {  
    $r = array();  
    $n = array();  
  
    // Get 4 semi-random numbers from 5 to 20 with  
    // a probability of being negative of 0.5  
    $n = getRandomArray(4, 5, 20, 5);  
  
    $q = "Expand and simplify ";  
    //      a      x  +  b  
    $temp = $n[0] * $n[1] + $n[2];  
    $q .= "$ (" . insertCoefficient($n[0], true, true) . insertCoefficient($n[1], false, false) . "y)(";  
    $q .= insertCoefficient($n[2], true, true) . insertCoefficient($n[3], false, false) . "y) $ ";  
    $q .= "<br />Give your answer like this <strong>ax^2 + bx + c</strong> where a, b and c are integers.";  
  
    $a = insertCoefficient(($n[1] * $n[3]), true, false) . "y^2";  
    $a .= insertCoefficient(($n[0] * $n[3] + $n[1] * $n[2]), false, false) . "y";  
    $a .= insertCoefficient($n[0] * $n[2], false, true);  
    array_push($r, '{"QN":"AL' . $inNum . ','"Q":"' . $q . '"}');  
    array_push($r, '{"QN":"AL' . $inNum . ','"Q":"' . $q . ','"A":"' . $a . ','"Type":2}');  
    return $r;  
}  
// =====
```

Fig. 4. The Code for generating question 2 on the Algebra test

The function returns a JSON object containing the question, the answer and a type variable which describes the type of answer. Answer types include:

- Text containing: the answer needs to contain a specific string or one of a number of specific strings
- Exact text: the answer needs to consist of an exact string like ax^2+bx+c
- Approximate number: The answer need to be within a specified tolerance of the actual answer
- Exact number: The number needs to be precisely correct

Fig. 5 shows questions rendered on the test page as they would be seen by a student during a test.

The screenshot shows a web interface for a test. At the top, there are navigation links: 'Baseline', 'Textbook', 'Sample Tests', 'Tests', 'My Progress', and 'Logout'. The main heading is '2 Algebra'. Below it are four questions:

1. Simplify the expression $(6y^3 + 10y^2 + 5y + 2) + (-12y^3 + 13y^2 + 14y + 1)$. Order the powers of y from the largest to the smallest.
2. Solve the equation $-13x + 19 = -111$. Give your answer as a number.
3. Expand and simplify $(6 + 13y)(-8 - 19y)$. Give your answer like this $ax^2 + bx + c$ where a, b and c are integers.
4. Evaluate $8(-16x + 18) + 17(-14x + 15) = 6(11x + 10)$.

Fig. 5. A test as seen by a student

3.4 Feedback

When using the sample tests students can press the enter key or the **Check My Answers** button to check their answers. Pressing the enter key causes the particular question to be marked. Clicking **Check My Answers** causes all questions with answers to be marked. The answers are marked in the browser and are not saved. This function helps students identify typographical mistakes as well as understand how to format answers. In Fig. 6 you can see correct and incorrect answers. If the student removes the minus sign and presses the enter key again the cross would change to a tick.

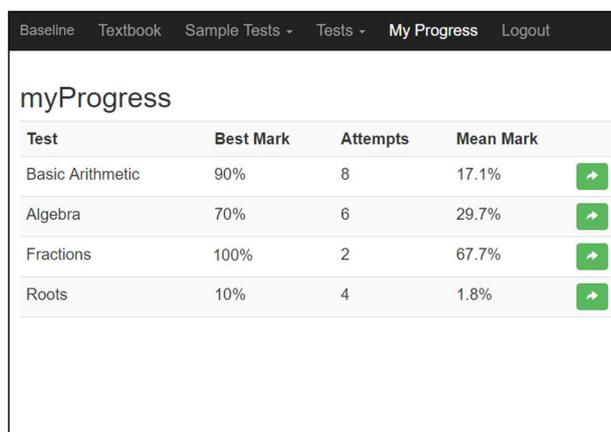
The screenshot shows the same test page as Fig. 5, but with feedback on the answers. The answers are:

- Question 1: $-11y^3+7y^2+2+7y$ (marked correct with a red checkmark)
- Question 2: -7 (marked incorrect with a red cross)
- Question 3: $-15y^2-8y+12$ (marked correct with a red checkmark)

 At the bottom left, there is a green button labeled 'Check My Answers'.

Fig. 6. Correct and incorrect sample test answers

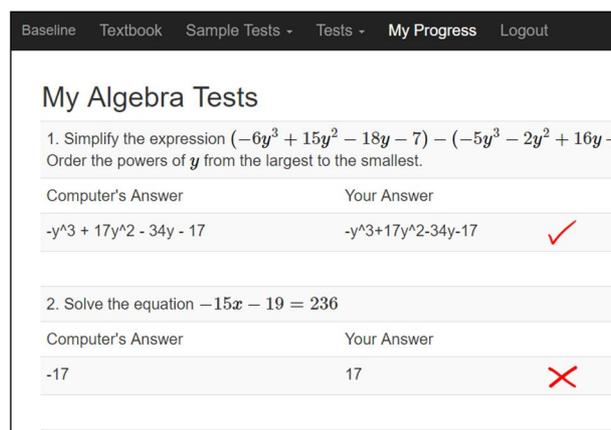
When a student sits one of the tests there is no feedback until the test is over. The questions and answers are saved and the results are displayed on the screen and listed in My Progress. See Fig. 7.



Test	Best Mark	Attempts	Mean Mark	
Basic Arithmetic	90%	8	17.1%	➔
Algebra	70%	6	29.7%	➔
Fractions	100%	2	67.7%	➔
Roots	10%	4	1.8%	➔

Fig. 7. Summary of a student's progress

Clicking on the green button on the right hand side of the screen opens a nested screen that contains details of each saved test. The screen shows each question, the expected answer, the student's answer and a tick or a cross to show whether the computer has found the answer to be right or wrong.



My Algebra Tests		
1. Simplify the expression $(-6y^3 + 15y^2 - 18y - 7) - (-5y^3 - 2y^2 + 16y + 7)$ Order the powers of y from the largest to the smallest.		
Computer's Answer	Your Answer	
$-y^3 + 17y^2 - 34y - 17$	$-y^3 + 17y^2 - 34y - 17$	✓
2. Solve the equation $-15x - 19 = 236$		
Computer's Answer	Your Answer	
-17	17	✗

Fig. 8. Typical feedback on an algebra test

3.5 Formatting

The mathematics displayed in Baseline are formatted using MathJax^[5]. One of the core goals of MathJax is high-quality display of mathematics notation in all browsers. That is also a requirement for Baseline. The output needs to be as good as possible. To output $ax^2 + bx + c$ in a browser we create the string `\$ ax^2 + bx + c \$` in the code that is sent to the browser. MathJax then searches for the `\$` markers and re-renders the content between the markers.

For students to input algebraic expressions for their answers the AsciiMath^[6] guidelines were adopted. For example, to enter $ax^2 + bx + c$ as an answer students would type `ax^2+bx+c`.

4. ANALYSIS OF THE PROJECT

When the baseline tests were carried out at the university a lot of work was required by the maths team at a time when they were busy getting students started on their new courses. Students with gaps in their knowledge generally did not seek help.

With Baseline students need to work on their own initiative and have a period during which they need to complete the tests.

Benefits to the student

1. Baseline is online which means students can access Baseline when they want – multiple times and at all hours
2. The tests make clear the minimum entry level mathematics expected and they cannot be avoided
3. Baseline is available in August before students have registered. It gets students thinking about studying in the year to come
4. Baseline imposes a precision both in arithmetic and algebra

Benefit to the course team

1. Provides a baseline metric of student ability
2. Almost no effort required on the part of the course team
3. Demands student engagement and sets an expectation before the start of teaching

5. STUDENT FEEDBACK

Questionnaires were sent to 230 students. We got 20 responses. Of the responses the key benefit was the availability 24 hours a day, 7 days a week. Students use Baseline on desktop computers, laptops and their phones.

6. FUTURE WORK

A number of short videos, up to 5 minutes long, were added to Baseline at the time students were revising for exams. Student response was immediate and positive. As a consequence the next release of Baseline will include a video dropdown in the top level menu. The videos will be stored in accordion folders with a video for each question on the sample papers.

In the longer term Baseline will benefit from a smarter marking function. To be useful the function would need to be able to parse a student's answer and mark it against the correct answer regardless of the format.

7. SOFTWARE AVAILABILITY

The software is available on Github and from Aston University.

Notes

- [1] Students are required to estimate values before they are calculated and then, when they have an answer, to ask whether the calculated value seems reasonable. Experience has shown that after two months the students routinely use a sanity check.
- [2] JSON: JavaScript object notation
- [3] LaTeX (pronounced latek) is a high quality typesetting system. LaTeX can be found here <http://latex-project.org/>
- [4] CSS: Cascading style sheets for formatting web pages
- [5] MathJax is a JavaScript library for formatting mathematical expressions in browsers. It is available from <https://www.mathjax.org/>
- [6] The AsciiMath website can be found at <http://asciimath.org/>