



# **Early findings from**

# Background

Isaac Computer Science (CS) is a **free, online platform supporting** students and teachers of A level Computer Science.

It is part of the National Centre for Computing Education, a Department for Education project aiming to boost computing education across England.

Isaac CS provides learning materials covering every topic of the A level Computer Science curriculum, as well as self-marking questions for students to practise with. The resources can be used in the **classroom** and for **independent study**, for **homework**, revision, extension activities, or flipped learning.

# **Bespoke features**

### Free-text questions

Free-text questions allow students to enter answers using natural language. The questions are then marked according to rules.

More on this in: Automated marking of free-text questions in STEM (Thomas & Beresford, 2020)

# **Boolean logic editor**

The Boolean logic editor is an interactive environment that allows students to construct a Boolean expression as the answer to a question.

More on this in: Effective use of mathematical equations in an online learning environment (Franceschini & Beresford, 2020)

# Formative assessment

Response systems (i.e. online platforms that auto-mark question can facilitate learning when they provide instant feedback that bo the students and the teacher can act upon (Kay & LeSage, 2009).

The 'markbook' feature of the platform provides teachers with a qui overview of their students' progress on a given assignment.

The 'My progress' feature shows students statistics about questions that they have answered and attempted.

### Engagement

Even though it is early to evaluate the impact of Isaac CS, data around user engagement suggests that Isaac CS has been positively received by the educational community.

#### **Events and competitions**

Isaac CS offers monthly competitions and a variety of training events for teachers and students that help them **learn** and **stay** connected with the educational community.

- Teacher CPD
- Student Masterclass • Discovery Day
  - Student Booster

#### References

Du Boulay, B. (1986) Some difficulties of learning to program. Journal of Educational Computing Research. 2 (1), 57–73. Guzdial, M. (2017) Balancing teaching CS efficiently with motivating students. *Communications of the ACM*. 60 (6), 10–11. Kay, R. H. & LeSage, A. (2009) Examining the benefits and challenges of using audience response systems: A review of the literature. Computers & Education. 53 (3), 819–827. Lister, R. et al. (2009) Further evidence of a relationship between explaining, tracing and writing skills in introductory programming. ACM SIGCSE Bulletin. 41 (3), 161–165. Parsons, D. & Haden, P. (2006) Parson's programming puzzles: a fun and effective learning tool for first programming courses. In: Tolhurst, D. and Mann, S. (eds.) Proceedings of the 8th Australasian Conference on Computing Education. Darlinghurst, Australia, Australian Computer Society, Inc. pp. 157–163. Sirkiä, T. & Sorva, J. (2012) Exploring programming misconceptions: an analysis of student mistakes in visual program simulation exercises. In: Proceedings of the 12th Koli Calling International Conference on Computing Education Research. New York, Association for Computing Machinery. pp. 19–28. Skudder, B. & Luxton-Reilly, A. (2014) Worked examples in computer science. In: Whalley, J. and D'Souza, D. (eds.) Proceedings of the Sixteenth Australasian Computing Education Conference. Darlinghurst, Australia, Australian Computer Society, Inc. pp. 59–64. Sorva, J. (2013) Notional machines and introductory programming education. ACM Transactions on Computing Education. 13 (2), 8:1–8:31.

2019-12-01 2020-01-01 2020-02-01 2020-03-01



	Content informed by research
Γ	Aisconceptions and
C	ommon wrong answers
A t a 1	well-established area of computing education research stresses he importance of identifying and resolving misconceptions fround programming (Sorva, 2013; Sirkiä & Sorva, 2012; Du Boulay, 986).
T r C ł	o address common misconceptions, we use carefully planned nultiple-choice, text, and numerical questions. Feedback for common wrong answers is provided to the student, which can help with resolving misunderstandings.
V f	Vrong answers entered into the platform are analysed monthly to ind misconceptions that we may not have anticipated.
Р	arson's Problems
sl a O P a	kills is Parson's Problems: answers are split into pieces, which re randomised, then the learner needs to put them in the right rder. arson's Problems are considered to be motivating (Guzdial, 2017) nd beneficial to the learner as they can be used to model rell-written code to encourage good programming practice
(F	Parsons & Haden, 2006).
(F Q a	Parsons & Haden, 2006). n Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. But the statements into the correct order (to match the logic of the regular
(F Q Q a	Parsons & Haden, 2006). n Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression).
(F Q Q a	Parsons & Haden, 2006). n Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items Your answer
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items a single character a Drag items across to build your answer
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items a single character a Zero or more instances of either a or b Drag items across to build your answer
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items a single character a zero or more instances of either a or b a followed by b creater area instances of either a or b
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items a single character a zero or more instances of either a or b a followed by b one or more instances of a zero or one instances of a
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items a single character a for or more instances of either a or b a followed by b one or more instances of a zero or one instances of a
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. Uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). <b>Available items</b> Vour answer a followed by b one or more instances of a tero or one instances of a Hint Hint2.
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. Usestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(alb)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). <b>Available items</b> Vour answer   a single character a   tere or more instances of either a or b   a followed by b   one or more instances of a   zero or one instances of a   thut   Hut1   Hut2
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. Usestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a7(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items Vour answer a single character a a followed by b one or more instances of a zero or one instances of a Intil Hint2 Check my answer
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. Usestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(alb)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). <b>Available items</b> <b>Sume Strings</b> <b>Sume Strings</b> <b>Sum </b>
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. ustions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Nuilable items string to the regular expression to build your answer string to remore instances of a build your answer Hint Hint2 Letek my answer teachers and students (state and
ngage	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a7(a b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items vour answer a single character a zero or more instances of either a or b a followed by b one or more instances of a zero or one instances of a build your answer thin1 Hint2 Check my answer bort forget to use the hints above if you need help. detechers and students (state and dent schools anywhere in the world) Gender ratio of engaged students
rgaged use	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. Usestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a)(b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression).  Available items Vour answer  isingle character a  isingle character
IVV (F O Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a)b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). <b>Available items</b> a single character a <u>rero or one instances of a time</u> Hint1 Hint2 <b>Check my answer</b> Drol leget to use the lows if you need help. <b>Bencher s and students (state and dent schools anywhere in the world)</b> rounts - students = teachers - total
VV (F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a7(alb)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). <b>Vour answer</b> Valiable items       Your answer            a single character a a rottowed by b a rot instances of a zero or one instances of a bitil Hint2             Mint1 Hint2           Dref grief use the birts above if you are help.
vv (F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(a)b)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items Vour answer a single character a rero or more instances of either a or b a followed by b ore or more instances of a zero or one instances of a detechers and students (state and dent schools anywhere in the world) rounds - students - teachers - total - students - teachers - total - students - teachers - total - students - teachers - total
VV (F Q Q a a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(alb)*aba*a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items a single character a a followed by b a followed by b b re or mere instances of either a or b a followed by b thin1 Hin2 Check my answer best regule use the bits above if you need help.
VV (F Q Q a a saged s s s s s s s s s s s s s s s s s s s	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(alb)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). <b>Vour answer</b> <b>Vour (</b>
(F Q Q a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression: a?(alb)*aba+a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items stopic character a rero or more instances of a iter a or b a followed by b one or more instances of a rero or one instances of a rero or one instances of a rero or one instances of a thin1 Hint2 Check my answer detechers and students (state and dent schools anywhere in the world) recurds - students - leachers - lotal - studen
VV (F Q Q a a so a so a	Parsons & Haden, 2006). In Isaac CS, Parson's Problems are used across multiple topics. uestions might involve creating a code snippet, ordering steps in process, or constructing a valid expression or sentence. The following statements describe the set of strings matched by the regular expression. a7(a b)*aba*a but they are not in the correct order. Put the statements into the correct order (to match the logic of the regular expression). Available items • single character = • single character

e array puzzle has the following values assigned to it: puzzle = [8, 9, 3, 5, 9, 3]
<pre>1 number = 9 2 answer = -1 3 for i = 0 T0 len(puzzle) - 1 4 if puzzle[i] == number then 5 answer = i 6 endif 7 next i What is the value of the variable answer once the code has finished executing? Value</pre>
Hint 1       Hint 2       Hint 3         Check my answer       Don't forget to use the hints above if you need help.

# Norked examples

he use of worked examples has been ighlighted as a key practice for computer cience because it enables students to evelop problem-solving techniques and nprove their performance in near transfer ests (Skudder & Luxton-Reilly, 2014).

ully worked examples are used to emonstrate an approach to solving a roblem, and to clarify difficult concepts. quivalent questions with three levels of ints function as faded worked examples to onsolidate learning.



#### Example of common wrong answers and feedback

**1** – This is a value that the variable **answer** takes, but it is not the final value.

**9** – This is the value of the variable **number**, not of the variable answer.

-1 – This is the initial value of **answer**, but not the final one.

**2** – The first element of the array is in position 0 (not 1). Also, the value **9** appears twice in the array.

**5** – Almost; the first element of the array is in position 0 (not 1).

# Tracing and reading code

Reading and explaining the purpo of code, as well as tracing co fosters the ability of novices to wr their own programs (Lister et 2009).

Isaac CS includes a variety questions that include tasks such asking students to fill in miss code, trace an algorithm to find output, or identify the over purpose of a snippet of code.

Where possible, this grad approach to learning has t extended to other parts of curriculum, for example, algorithn

Each question is accompanied b video hint that provides a par solution or some of the working, but never the full answer.



ose ode, rite	Travis is working with some data stored in CSV format. Each line of data contains multiple data values separated from each other by commas. Travis has written a recursive procedure to help him output a line of data in a more readable way.
al.,	<pre>procedure csv_output(whole_string, delimiter)     location = position(whole_string, delimiter)     word = whole_string.subString(0,location)     rest = whole string.subString(location+1, whole string.length-(</pre>
of n as sing	<pre>if location &gt;= 0 then     print(word)     csv_output(rest, delimiter)     else         print(rest)     endif endprocedure</pre>
rall	Remember that the subString function takes two arguments: the first character to include and the number of characters required. Assume that each print statement displays on a new line.
dual een the ns.	If Travis calls this procedure with the following arguments, what will the output be? csv_output("do,you,know,the,muffin,man", ",") Type your answer here.
oy a rtial	Hint 1 Hint 2 Hint 3 Check my answer

Don't forget to use the hints above if you need hel

Moving forwards, there is scope to analyse the various learning paths that students follow, pinpoint challenging topics, and test interactive features to understand how to best support learners.

An annual survey of the registered users will be carried out to help evaluate the impact of the programme.

Even though the percentage of active female students reflects the gender balance in computer science education, it is an area