Online assessment platform enables different levels of question randomizations

Cinda Heeren, Stephan Koenig, Firas Moosvi and Steven Wolfman

Based on workshop material by Firas Moosvi

2023-05-01

PrairieLearn, an online assessment tool

- Facilitates rich and unconstrained question types
- Supports isomorphic questions and randomization
- Offers questions in different syllabus-driven contexts:
  - Practice problems incentivize repetition and enable mastery learning
  - Formative engagement, like projects, requires little randomization
  - Pre-class quizzes and in-class group work support flipped classrooms
  - Short exams that are more frequent at lower stakes and with second chances

Standard

Standard questions with no randomization

Abdallah drops a 3 kg ball from the top of a 20 m tall building. How long does the ball take to hit the ground?

Surface Features

Change surface level features such as names, colours, phrases, objects, etc…

Increased cognitive load for students when they are pattern-matching.

{{ Abdallah }} drops a 3 kg {{ ball }} from the top of a 20 m tall {{ building }}. How long does the {{ ball }} take to hit the ground?

BTree Motivation

Select each statement that is true about BTrees of order m, and height c, containing s keys. In the statements below, we use the phrase "expensive data operations" to denote any kind of remote data transfer required by an algorithm, including things like disk seeks and api calls. All running times are considered to be worst case.

- [ ] Searching for a key within a node invokes several expensive data operations.
- [ ] The worst-case time spent searching for a key within a node is $O(m)$.
- [ ] As s increases sufficiently for a particular BTree, m will also automatically increase.
- [ ] For a fixed value of s, if a single node still will not "use up" one expensive data operation, we can decrease the number of expensive data operations by decreasing m.

Select all possible options that apply.

Save & Grade | Save only | New variant

BTree Motivation

Select each statement that is true about BTrees of order b, and height d, containing k keys. In the statements below, we use the phrase "expensive data operations" to denote any kind of remote data transfer required by an algorithm, including things like disk seeks and api calls. All running times are considered to be worst case.

- [ ] The worst-case time spent searching for a key within a node is $\Theta(d)$.
- [ ] The worst-case time spent searching for a key within a node is $\Omega(\log b)$.
- [ ] In practice, d is much larger than b.
- [ ] d is an asymptotic upper bound on the number of expensive data operations.

Select all possible options that apply.

Save & Grade | Save only | New variant

Conditions

Change problem conditions and/or values.

Repeated retrieval practice so students can do their own work even in group settings.

Abdallah drops a {{ 3 kg }} ball from the top of a {{ 20 m }} tall building. How long does the ball take to hit the ground?

Pointers with Numbers: Traversing a List

Nodes in a number-pointer linked list will be from this `struct`, where a value of `-1` acts like a null pointer:

```
struct Node
{
    char data;
    int next;
};
```

The number-pointers refer to elements we have added to the `memory` vector: `vector<Node> memory;`.
Now, consider the following contents of `memory`:

	0	1	2	3	4	5	6	7	8	9
data	o	b	a	s	k	j	n	i	o	l
next	0	8	1	-1	7	4	0	-1	4	8

What word is stored in the linked list whose head is at number-pointer `2`? (Include just the letters in the `data` fields of the linked list, with no extra text like quotation marks, spaces, or commas.)

Linked list word

Save & Grade | Save only | New variant

Pointers with Numbers: Traversing a List

Nodes in a number-pointer linked list will be from this `struct`, where a value of `-1` acts like a null pointer:

```
struct Node
{
    char data;
    int next;
};
```

The number-pointers refer to elements we have added to the `memory` vector: `vector<Node> memory;`.
Now, consider the following contents of `memory`:

	0	1	2	3	4	5	6	7	8	9
data	q	o	a	i	s	i	n	w	d	a
next	-1	5	8	0	2	8	6	3	3	7

What word is stored in the linked list whose head is at number-pointer `4`? (Include just the letters in the `data` fields of the linked list, with no extra text like quotation marks, spaces, or commas.)

Linked list word

Save & Grade | Save only | New variant

Contexts or Scenarios

Assess the same concept from a different perspective by changing the problem context or scenario.

Intentional pattern-matching so students can synthesize patterns to learn concepts and understand how to apply them.

Abdallah {{ throws a ball up }} and it travels a height of 20 m {{ until it stops }}. How long does the ball take to hit the ground?

Linked List Lab Examlet Follow-Up

Consider this `Node` structure:

```
struct Node {
  int data;
  Node *next, *prev;
  Node(int data_, Node *next_ = nullptr, Node *prev_ = nullptr) :
    data(data_), next(next_), prev(prev_) {}
};
```

Complete the function `ping(Node *tail, int p)` below that operates on a doubly-linked list of these `Node`s, accessed via a pointer to its **last** node. (That pointer is null if the list is empty.)

The function:

1. Finds the `p`th node (where `p >= 0`)—which we'll call `target`—by traversing **backward** from the `tail`. For example, if the list contained `[-23, 33, -22, -37]` and `p == 2`, the node found would be the one containing `33`. (If `p == 0`, it stops immediately.)
2. Then traverses **forward** through the list from `target`—including inspecting `target` itself—and finds the first node containing a negative number. For example, if the list contained `[-23, 33, -22, -37]` and `p == 2`, the node found would be the one containing `-22`. (If the first number checked is negative, it stops immediately.)
3. Returns a pointer to the node on which the previous step finishes.

Linked List Lab Examlet Follow-Up

Consider this `Node` structure:

```
struct Node {
  int data;
  Node *next, *prev;
  Node(int data_, Node *next_ = nullptr, Node *prev_ = nullptr) :
    data(data_), next(next_), prev(prev_) {}
};
```

Complete the function `mest(Node *head, int q)` below that operates on a doubly-linked list of these `Node`s, accessed via a pointer to its first node. (That pointer is null if the list is empty.)

The function:

1. Finds the first node containing a number with an absolute value less than `100` (the `abs` function may be helpful)—which we'll call `target`—by traversing forward from the `head`. For example, if the list contained `[125, -114, 35, -37]`, the node found would be the one containing `35`. (If the first number checked has an absolute value less than `100`, it stops immediately.)
2. Then traverses **backward** through the list from `target`—including inspecting `target` itself—and finds the `q`th node (where `q >= 0`). For example, if the list contained `[125, -114, 35, -37]` and `q == 1`, the node found would be the one containing `-114`. (If `q == 0`, it stops immediately.)
3. Returns a pointer to the node on which the previous step finishes.

Concepts (or Principles)

Assess different concepts in a particular context or perspective.

Students need to understand concepts deeply and cannot pattern-match efficiently.

Abdallah drops a 3 kg ball from the top of a 20 m tall building. What is its {{ velocity }} just before it hits the ground?

Data types? Huh. Yeah. What are they good for?

Consider the following data type definition:

```
data PingGizmo = Bar Integer
               | XyzzyMell Integer Ham Bool
               | Plugh String
               | IpsumFitch
```

Select **all** of the following that this defines:

- [ ] (a) A data constructor named `IpsumFitch`
- [ ] (b) A function named `Corge`
- [ ] (c) A type named `Ham`

Select all possible options that apply.

Select **all** of these that we can do once we have this definition:

- [ ] (a) Pattern match with a function definition like: `h (PingGizmo _ _ _) = ...`
- [ ] (b) Pattern match with a function definition like: `h IpsumFitch = ...`
- [ ] (c) Pattern match with a function definition like: `h (XyzzyMell _ _ _) = ...`

Select all possible options that apply.

Save & Grade | Save only | New variant

Data types? Huh. Yeah. What are they good for?

Consider the following data type definition:

```
data Gonk = Lorem
          | Quux Integer String
          | Mell CorgeIpsum Bool
```

Select **all** of the following that this defines:

- [ ] (a) A type named `Gonk`
- [ ] (b) A data constructor named `Mell`
- [ ] (c) A type named `Lorem`

Select all possible options that apply.

Select **all** of these that we can do once we have this definition:

- [ ] (a) Define a function with the type `Mell -> Mell`
- [ ] (b) Pattern match with a function definition like: `f (Mell _ _) = ...`
- [ ] (c) Define a function with the type `Gonk -> Gonk`

Select all possible options that apply.

Save & Grade | Save only | New variant

Different question

Completely different question about the same or related concept.

Allows for multiple attempts of assessments for repeat (or, second-chance) testing.

{{ Abdallah drops a 3 kg ball, a 5 kg box, and a 1 kg backpack from the top of a 20 m tall building. Rank the items in the order they hit the ground. }}

Keys

Each problem below is a brief scenario followed by an answer box. You must determine how to use a dictionary to address the scenario and then describe what an arbitrary key in the dictionary would be. Each description is **a word or a phrase of two consecutive words, directly taken from the text of the scenario**. (All answers are in singular form, and capitalization is ignored.)

Example:

> You are given a list of the students in CPSC 221 and the number of Piazza posts each answered during the semester. We want to know which students answered more than 100 Piazza posts this semester.
>
> key description: student

---

1. Every student has chosen one Zoom call to join with their friends. Determine the number of students in any given Zoom call.

   key description:

2. Assume you are given a tree and seven colors. You want to color each node so its color is different from any adjacent node (parent or child). How would you keep track of the coloring choices?

   key description:

3. The coop office has a database containing each student, together with their company. Determine the list of students employed by each company.

   key description:

Save & Grade | Save only | New variant

Keys

Each problem below is a brief scenario followed by an answer box. You must determine how to use a dictionary to address the scenario and then describe what an arbitrary key in the dictionary would be. Each description is **a word or a phrase of two consecutive words, directly taken from the text of the scenario**. (All answers are in singular form, and capitalization is ignored.)

Example:

> You are given a list of the students in CPSC 221 and the number of Piazza posts each answered during the semester. We want to know which students answered more than 100 Piazza posts this semester.
>
> key description: student

1. Every course has a different enrollment size. We want to identify courses with over 300 students!

   key description:

2. You are given a collection of itemized receipts from a grocery store, and a list of scarce products (e.g. toilet paper). You want to know what other products people typically buy when they buy each scarce product.

   key description:

3. Every day in the month of April has an associated temperature. You want to know the temperature on the 15th day.

   key description:

Save & Grade | Save only | New variant

Discussion

- What valuable pedagogical approaches does randomization enable?
- What pedagogical problems arise from randomization, and how can we minimize them?
- Which kinds of randomization meet different teaching and learning goals?