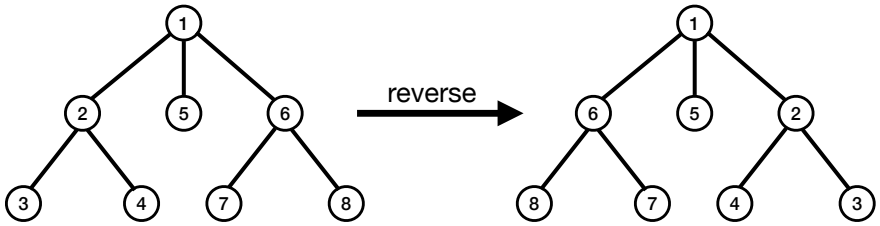
Name:

BUID:

Workshop 2: Practices with Tuples, Records, ADTs

1. Implement the function `val reverse : 'a tree -> 'a tree` which reverses a tree.



2. Implement the function `val get_int : string -> (int * string) option` so that `(get_int s)` is

   - `Some (n, rest)` if `s` starts with a string that represents an integer, and `rest` is the remaining part of the string.
   - `None` if `s` does not begin with an integer representation.

3. *(Challenge)* Implement the function `val eval : expr -> (int, error) result` so that `eval e` is

   - `Ok n` if `e` is a valid expression with the value `n`;
   - `Error DivByZero` if `e` contains a division by zero;
   - `Error NegExp` if `e` contains a negative exponent.

   If an expression has multiple errors, `eval e` should report the *leftmost* error.

4. *(Challenge)* Implement the function `val get_attachment : content_type -> 'a email -> 'a list` so that the expression `get_attachment ct e` is the list of attachments in `e` whose headers say their content type is `ct` (first take a look at the types and make sure you understand how they work).

Write down below who you worked with and which problems you finished.