This assignment uses polymorphism. You will build a small class hierarchy for simple containers of integers. Specifically, you must build the following objects.

1. **(10 points)** An abstract base class, intbox, with constructors and destructors as necessary, and containing two pure virtual methods that must be provided in all derived classes:
   * Constant method bool contains(int a), which returns true if the container holds integer a, and false otherwise.
   * Constant method void show(ostream &s), which displays the container onto stream s (see below for details).
2. **(10 points)** Class singleton, derived from intbox, representing a single integer and containing:
   * An integer value.
   * A constructor that takes an integer, for setting value.
   * A destructor if necessary.
   * Method contains, returning true iff the values match.
   * Method show, which simply outputs value.

As an example, the singleton for value 6 would be displayed as 6 and contains only the integer 6.

1. **(10 points)** Class interval, derived from intbox, representing an interval (all integers between two endpoints) and containing:
   * Integers low and high, the endpoints of the interval.
   * A constructor that takes the lower and upper bounds for the interval and sets low and high appropriately. Throw an error if low is greater than high.
   * A destructor if necessary.
   * Method contains, returning true iff the value is contained in the interval.
   * Method show, which should display [low, high].

As an example, the interval with low=1 and high=5 would be displayed as [1, 5] and contains the integers 1, 2, 3, 4, 5.

1. **(20 points)** Class collection, derived from intbox, to represent a collection of other containers, containing:
   * An unsigned integer size, and an array (of dimension size) of pointers to intbox.
   * A constructor that takes an unsigned integer, and sets its size and allocates the array accordingly. The array elements should be initialized to null pointers.
   * A destructor if necessary.
   * Method void set\_item(unsigned i, intbox\* item) for setting element i of the array.
   * Method contains, returning true iff some array element contains the value.
   * Method show, which should display the array elements as {item[0], item[1], ..., item[size-1]} where each item is displayed in its appropriate way.

As an example, the collection of size 3 whose first item is the singleton 6, whose second item is a collection of size 2 containing singletons 9 and 13, and whose third item is the interval [1,5], would be displayed as {6, {9, 13}, [1, 5]} and contains the integers 1, 2, 3, 4, 5, 6, 9, 13.

Class definitions should go in intbox.hh, and all implementation should go in intbox.cc. To test your implementation, you may use main.cc (see the git repository for class) which includes intbox.hh and reads an input file. The input file builds a container and then contains queries. Example inputs, and their corresponding outputs, are given in the git repository. Submit your source files, intbox.hh and intbox.cc, in Canvas.