Source File:	`~/2336/23/lab23.cpp`
Input:	under control of `main` function
Output:	under control of `main` function
Value:	2

Write a recursive function template whose prototype is given by

```
template<typename T>
int numDistinct(const T *array, int n);
```

The function returns the number of distinct elements in the range `[array, array + n)`. A `main` function for testing your function is shown in Figure 1. The expected output from executing this code is shown in Figure 2. To use the `Makefile` as distributed in class, add a target of `lab23main` to `targets1srcfile`.

```
#include <iostream>
#include <string>

using namespace std;

// function template prototype
template<typename T>
int numDistinct(const T *array, int n);

#include "lab23.cpp"

template<typename T>
void printArray(const T *array, int count)
{
  if (count > 0)
  {
    cout << *array << " ";
    printArray(array + 1, count - 1);
  }
  else
    cout << endl;
}

template<typename T>
void printAndCountDistinct(const T *array, int n, string nameOfArray)
{
  int distinct;

  cout << "Array " << nameOfArray << " contains:" << endl;
  printArray(array, n);
  distinct = numDistinct(array, n);
  cout << "and has " << distinct << " distinct element"
       << (distinct > 1 ? "s." : ".") << endl << endl;
}

```

Figure 1. `/usr/local/2336/src/lab23main.C` (Part 1 of 2)

```
int main()
{
  const int aCount = 5, bCount = 7, cCount = 7, dCount = 12;
  int a[aCount] = {5, 5, 5, 5, 5};
  double b[bCount] = {7.7, 6.6, 5.5, 4.4, 3.3, 2.2, 1.1};
  char c[cCount] = {’r’, ’a’, ’c’, ’e’, ’c’, ’a’, ’r’};
  string d[dCount] = {"Cadillac", "Oldsmobile", "Chevrolet",
                      "Toyota", "Lexus", "Dodge", "GMC", "BMW",
                      "BMW", "GMC", "Dodge", "Lexus"};

  printAndCountDistinct(a, aCount, "a");
  printAndCountDistinct(b, bCount, "b");
  printAndCountDistinct(c, cCount, "c");
  printAndCountDistinct(d, dCount, "d");

  return 0;
}
```

Figure 1. `/usr/local/2336/src/lab23main.C` (Part 2 of 2)

```
newuser@csunix ~> cd 2336
newuser@csunix ~/2336> ./getlab.ksh 23
  * Checking to see if a folder exists for Lab 23. . .No
  * Creating a folder for Lab 23
  * Checking to see if Lab 23 has sample input and output files. . .Yes
  * Copying input and output files for Lab 23
    from folder /usr/local/2336/data/23 to folder ./23
  * Checking to see if /usr/local/2336/src/lab23main.C exists. . .Yes
  * Copying file /usr/local/2336/src/lab23main.C to folder ./23
  * Checking to see if /usr/local/2336/include/lab23.h exists. . .No
  * Copying file /usr/local/2336/src/Makefile to folder ./23
  * Adding a target of lab23main to targets1srcfile
  * Touching file ./23/lab23.cpp
  * Edit file ./23/lab23.cpp in Notepad++
newuser@csunix ~/2336> cd 23
newuser@csunix ~/2336/23> ls
01.dat       01.out       Makefile     lab23.cpp    lab23main.C
newuser@csunix ~/2336/23> make lab23main
g++ -g -Wall -std=c++11 -c lab23main.C -I/usr/local/2336/include -I.
g++ -o lab23main lab23main.o -L/usr/local/2336/lib -lm -lbits
newuser@csunix ~/2336/23> ./lab23main
Array a contains:
5 5 5 5 5
and has 1 distinct element.

Array b contains:
7.7 6.6 5.5 4.4 3.3 2.2 1.1
and has 7 distinct elements.

Array c contains:
r a c e c a r
and has 4 distinct elements.

Array d contains:
Cadillac Oldsmobile Chevrolet Toyota Lexus Dodge GMC BMW BMW GMC Dodge Lexus
and has 8 distinct elements.

newuser@csunix ~/2336/23> ./lab23main > my.out
newuser@csunix ~/2336/23> diff 01.out my.out
newuser@csunix ~/2336/23>
```

Figure 2. Commands to Compile, Link, & Run Lab 23