CS111 Homework 5 (Due Nov 29, 2018)

Name your program **hw5a.cpp** for part A and **hw5b.cpp** for part B. Program must be able to **compile** or you will get at most 0.5 out of 2 points for the assignment.

Run the following command in venus to copy the starter code:

```
cp ~ctse/cs111/hw5* ~/hwSubmission/
```

Part A: Recursion (Absolute C++ textbook ch.13)
Use the starter code hw5a.cpp, modify the program by writing the recursive functions. Do not modify the main function.

Question 8. Interest

A savings account typically accrues savings using compound interest. If you deposit $1,000 with a 10% interest rate per year then after one year you have $1100. If you leave this money in the account for another year at 10% interest then after two years you have $1,210. After three years you would have $1,331, and so on.

Write a program that inputs the amount of money to save, an interest rate per year, and the number of years the money will accrue compound interest. Write a recursive function that calculates the amount of money that will be in the savings account using the input information.

To verify your function, the amount should be equal to $P(1+i)^n$ where P is the amount initially saved, i is the interest rate per year, and n is the number of years.

Question 10. Bowling Pins

Consider a frame of bowling pins shown below, where each * represents a pin:

```
     *
    * *
   * * *
  * * * *
* * * * *
```

There are five rows and a total of fifteen pins.
If we had only the top four rows then there would be a total of 10 pins.
If we had only the top three rows then there would be a total of 6 pins.
If we had only the top two rows then there would be a total of 3 pins.
If we had only the top row then there would be a total of 1 pin.

Write a recursive function that takes as input the number of rows n and outputs the total number of pins that would exist in a pyramid with n rows. Your program should allow for values of n that are larger than 5.

Part B: (from professor Michael Fried)
Use the starter code hw5b.cpp, modify the program by writing the following functions. Do not modify the main function.

1. Write a **range** function:
    - This function should take 2 parameters: a 1D int array and the number of elements.
    - It should return the difference between the largest element and the smallest element.
2. Write a **reverse** function:
    - This function should take 2 parameters: a 1D int array and the number of elements.
    - It should reverse the order of the elements. It should not return anything.
    - Hint: You can use the swap function to swap two elements, for example swap(a[i], a[j]) would swap a[i] and a[j].

Sample run:

```
The range of array a is 6
The range of array b is 4
Array a reversed: 3 8 2 7
Array b reversed: 7 6 5 4 3
```