Submit a writeup of your design, screenshots of the results, and code files in a zip file.

A screenshot of a computer

Description automatically generated

**Figure 1.** Console output from demo program employing the use of the Team class, which uses an array-based stack in a double-ended fashion.

The implementation of the Team ADT uses an array-based stack under the hood. It is assumed that the user of the data structure initializes the capacity to account for the total team size, including both junior varsity and varsity team members. The underlying array contains two stacks: one which begins with the lowest array index, and one which begins with the highest array index. The two stacks grow in size approaching the center index of the array as members are added to a particular team. The implementation details of the varsity stack are straightforward and follow that of a traditional array-based stack. The junior varsity stack of team members originates from the end of the array and its methods are implemented in a reverse fashion, such that the first added member will exist at the highest index of the array, and the next added member will exist at the array index one less than this. Popping from the latter stack will increment the index pointer and cause the stack top element to be one index nearer to the end of the array.