In-class Exercise 8: Vectors

For this exercise, you may work with another student. If you do, turn in **ONLY one copy** of the project with both names on it. Turn in your project at the end of class and name it as follows:

- If you have not completed the whole implementation and the comments, name your file:
  A250_TEMP_E8_Yourlastname_Yourfirstname
  (or **A250_TEMP_E8_Yourlastname_Yourfirstname_Otherstudentlastname_Otherstudentfirstname**)
  The complete project is due **next week on Wednesday/Thursday**.

- If you have completed the entire exercise, name your file:
  A250_E8_Yourlastname_Yourfirstname
  (or **A250 _E8_Yourlastname_Yourfirstname_Otherstudentlastname_Otherstudentfirstname**)

In an ancient land, the beautiful princess Eve had many suitors. She decided on the following procedure to determine which suitor she would marry. First, all of the suitors would be lined up one after the other and assigned numbers. The first suitor would be number 1, the second number 2, and so on up to the last suitor, number n. Starting at the first suitor she would then count three suitors down the line (because of the three letters in her name) and the third suitor would be eliminated from winning her hand and removed from the line. Eve would then continue, counting three more suitors, and eliminating every third suitor. When she reached the end of the line she would continue counting from the beginning.

For example, if there were **6 suitors** then the elimination process would proceed as follows:

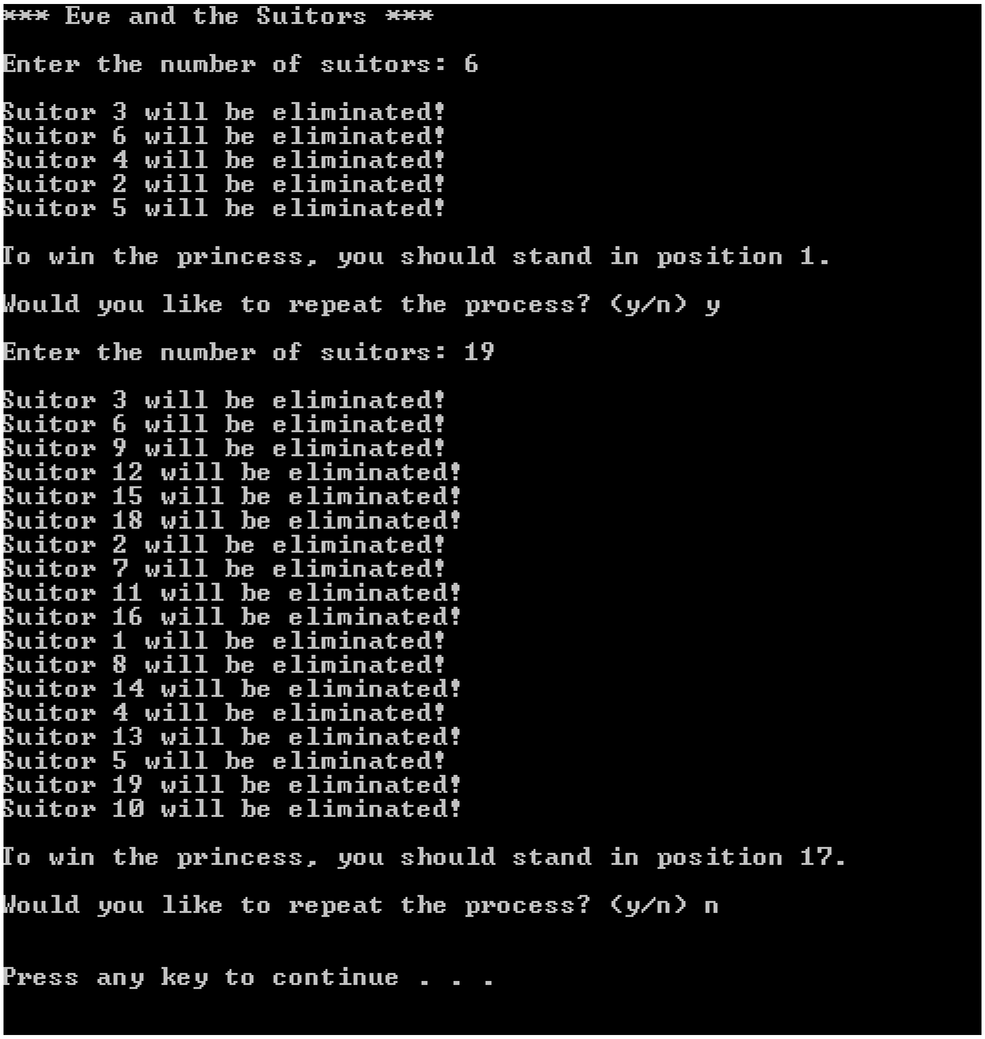
```
123456 initial list of suitors, start counting from 1
12456  suitor 3 eliminated, continue counting from 4
1245   suitor 6 eliminated, continue counting from 1
125    suitor 4 eliminated, continue counting from 5
15     suitor 2 eliminated, continue counting from 5
suitor 5 eliminated, 1 is the lucky winner
```

Write a program that uses an **STL vector** to determine which position a suitor should stand in to marry the princess if there are n suitors. Allow the user to repeat the process.

You are allowed to:

- Create **ONLY one (1) STL vector** and NO additional containers (not even an array)
- Create **ONLY three (3)** variables: **two (2)** of type **int** and **one (1)** of type **char**
- Create **one (1) iterator** (**Note:** the function **erase** returns an iterator)
- Use **ONLY** the following **STL vector functions**:
  - size
  - erase
  - begin
  - end
  - overloaded constructor (passes the number of suitors)
  - overloaded subscript operator [ ]

POSSIBLE OUTPUT <= By “POSSIBLE,” I am referring to the format only; the results should be the same.



```
*** Eve and the Suitors ***

Enter the number of suitors: 6

Suitor 3 will be eliminated!
Suitor 6 will be eliminated!
Suitor 4 will be eliminated!
Suitor 2 will be eliminated!
Suitor 5 will be eliminated!

To win the princess, you should stand in position 1.

Would you like to repeat the process? (y/n) y

Enter the number of suitors: 19

Suitor 3 will be eliminated!
Suitor 6 will be eliminated!
Suitor 9 will be eliminated!
Suitor 12 will be eliminated!
Suitor 15 will be eliminated!
Suitor 18 will be eliminated!
Suitor 2 will be eliminated!
Suitor 7 will be eliminated!
Suitor 11 will be eliminated!
Suitor 16 will be eliminated!
Suitor 1 will be eliminated!
Suitor 8 will be eliminated!
Suitor 14 will be eliminated!
Suitor 4 will be eliminated!
Suitor 13 will be eliminated!
Suitor 5 will be eliminated!
Suitor 19 will be eliminated!
Suitor 10 will be eliminated!

To win the princess, you should stand in position 17.

Would you like to repeat the process? (y/n) n


Press any key to continue . . .
```