Assignment #5
Practice Using Arrays
Due: Sunday, 11/19/17, 11:59pm

(90 pts) **Problem Statement:** Write a C++ program that plays the game of Yahtzee. Here are instructions for playing the game:
http://www.hasbro.com/common/instruct/Yahtzee.pdf
You can also find a mobile version to play in the Google play store or App store.

Your game will allow 1-2 players, and at the end, you need to ask if the user wants to play again and with how many players. There are 5 six-sided dice in the game and you get three rolls to try to make something on the scoresheet. You must use an array to hold the values of the 5 dice and an array to hold the scores for the 16 categories for a player.

The top half of the scoresheet is only a sum of 1 or more of dice you are trying to get: 1s, 2s, 3s, 4s, 5s, and 6s, and if you score 63 or more points in this section, you get an additional 35 points.

The bottom half of the scoresheet consist of options that have pre-defined points or a total of all the dice. You have the options for a three of a kind (total of all dice, but has to have 3 or more of the same number), four of a kind (total of all dice, but has to have 4 or more of the same number), full house (25 pts, 2 dice the same and 3 dice the same), small straight (30 pts, 4 numbers in a row), large straight (40 pts, 5 numbers in a row), Yahtzee (50 pts, 5 dice the same number), up to 3 bonus Yahtzees (100 pts each, 5 dice the same number), and chance (total of all dice, any combination).

After your first roll, a player can choose to roll 0-5 specific dice again. The player needs to state how many dice they want to re-roll and which ones they want to re-roll by providing the array indices for each die. If the player chooses not to roll again or has rolled 3 times, then the player must choose a category not previously chosen on the scoresheet. If the dice do not match the rules for that category, then the player takes a 0 for that category. Once a category has a score (even a zero score), it cannot be chosen again, so think carefully about what value you are going to use to initialize the scoresheet!!!

The bonus Yahtzee can only be chosen if there is a non-zero score in the Yahtzee!!! The player's turn is over once a category is chosen, and the game is over once all players have put a score in all categories. The final score is a total points from the two sections, including the bonus if the top section is 63 or greater.

Command-line argument
To support various numbers of players, you will run the program with the specific number at runtime, rather than prompting for the input. The user can enter the number of players, 1-2. Any other command line argument is an error.

1-player Example:
./yahtzee 1

Your program must be able to:

- Print an error message and recover, when the player doesn't supply a valid category. This includes selecting a category with a score, which can include a zero for the score. Choose your initialize value for the scoresheet carefully!
- Print an error message when the user enters an invalid option as a command line argument. You do not need to recover from this.
- Correctly determine the score for the category based on the dice (Remember, if the dice do not match the rules for the category, then a zero score is placed in that category).
- Play the game correctly based on rules and number of players.
- Continue to play until the user selects no.
- Your functions are 15 lines or less
- You must not have any global variables
- You must use a dynamic 1-d array for the second player's scoresheet, since there may or may not be a second player.
- You must not have memory leaks.

(10 pts Extra Credit) Yahtzee Any Number Players

Instead of having only 1-2 players, your Yahtzee will support any number of players using a dynamic 2-d array. This would have 16 rows/categories by N players.

(10 pts) Program Style/Comments

In your implementation, make sure that you include a program header in your program, in addition to proper indentation/spacing and other comments! Below is an example header to include. Make sure you review the style guidelines for this class, and begin trying to follow them, i.e. don't align everything on the left or put everything on one line! http://classes.engr.oregonstate.edu/eecs/fall2017/cs161-001/161_style_guideline.pdf

```
/***************************************************
** Program: yahtzee.cpp
** Author: Your Name
** Date: 11/15/2017
** Description:
** Input:
** Output:
***************************************************/
```

Electronically submit your **C++ program** (**.cpp** file, not your executable!!!) by the assignment due date, using TEACH.