Classes: A First Look

```
#include <iostream.h>

#define SIZE 10

// Declare a stack class for characters
class stack {
    char stck[SIZE]; // holds the stack
    int tos;          // index of top-of-stack

public:
    void init();          // initialize stack
    void push(char ch); // push character on stack
    char pop();           // pop character from stack
}
```

```
// Initialize the stack
void stack::init() { tos = 0; }

// Push a character.
void stack::push(char ch) {
    if (tos==SIZE) { cout << “Stack if full”; return; }
    stck[tos] = ch;
    tos++; }

// Pop a character
char stack::pop() {
    if (tos==0) { cout << “Stack is empty”;
                  return 0; // return null on empty stack
                }
    tos--; return stck[tos]; }
```

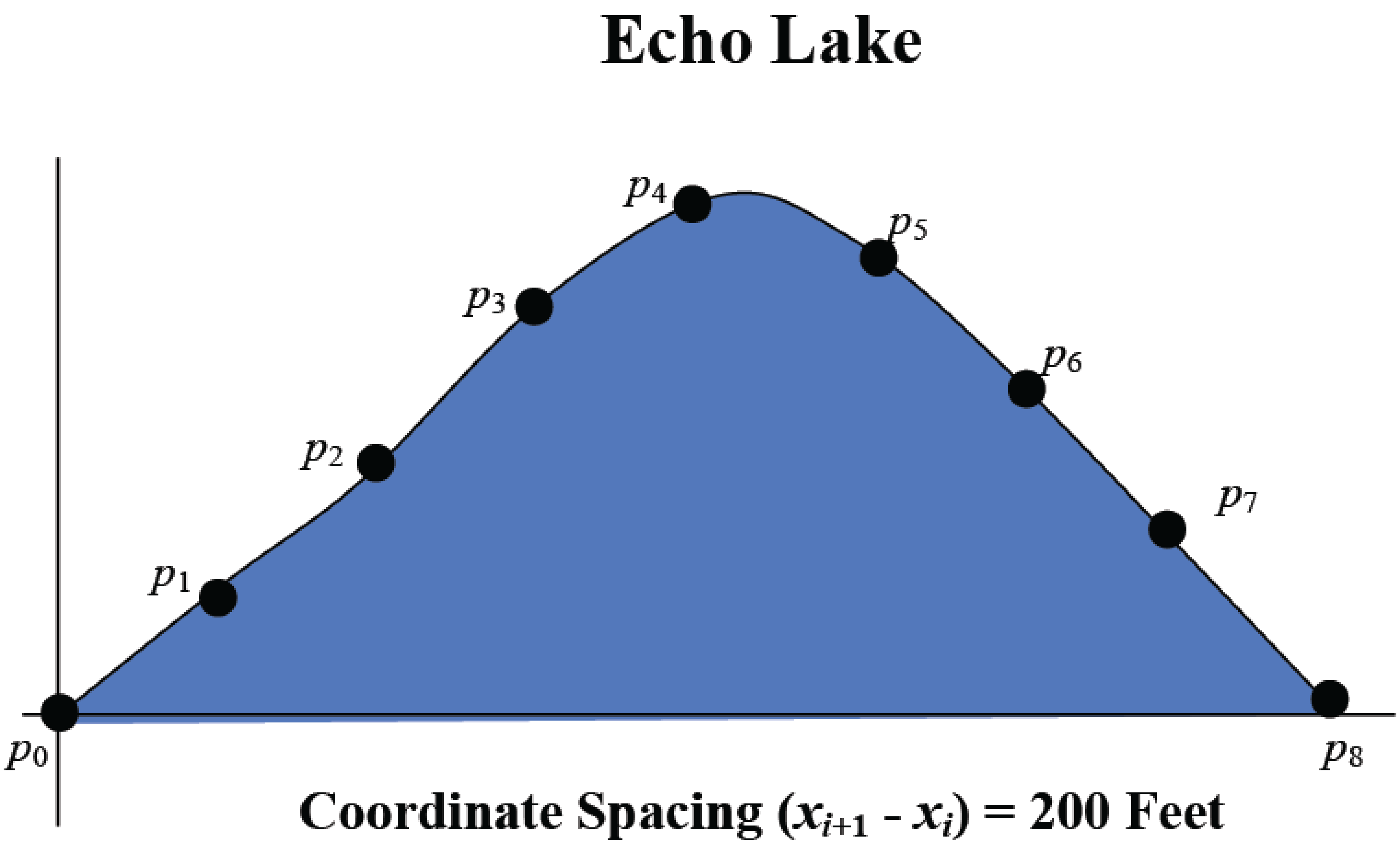
```
main() {
    stack s1, s2; // create two stacks
    int i;
    // initialize the stacks
    s1.init();
    s2.init();

    s1.push(‘a);               s2.push(‘x’);
    s1.push(‘b’);              s2.push(‘y’);
    s1.push(‘c’);              s2.push(‘z’);

    for (i=0; i<3; i++) cout << “Pop s1: “ << s1.pop() << “\n”;
    for (i=0; i<3; i++) cout << “Pop s2: “ << s2.pop() << “\n”;

    return 0;
}
```

This page intentionally left blank

HW #2 (Stock a lake with fish)

- Design a modular C++ (or Java or Python if you prefer) program to solve the following problem.

- You are the game warden狩獵監督官 in your town and are responsible for stocking the local lake (see Figure 1) prior to the opening of the fishing season. The average depth of the lake is 20 feet. Your plan is to stock the lake with 1 fish per 1000 cubic feet, and have approximately 25% of the original fish population remaining at the end of the season. What is the maximum number of fishing licenses that can be sold if the average catch is 20 fish per license?



Echo Lake
p_4
p_5
p_3
p_6
p_2
p_7
p_1
p_0
p_8
Coordinate Spacing $(x_{i+1} - x_i)$ = 200 Feet

Figure 1

HW #2 (2)

Input

- A data file containing space-delimited integers representing the *y* values of the coordinate points in Figure 1. Note that the file may or may not have a **newline** character before **EOF**.

Output

- The annotated, maximum number of fishing licenses that can be sold if the average catch is 20 fish per license.

HW #2 (3)

Hint

- You will have to compute the volume of the lake, since the criterion for stocking is 1 fish per 1000 ft^3. Volume can be computed as *Area* x *Depth*. The area can be computed using calculus. In particular, use *Simpson's Rule* to compute the area. Recall we can compute the area under a curve using Simpson's Rule as follows.

$$\int_a^b f(x)dx \approx \frac{h}{3}(y_0 + 4y_1 + 2y_2 + 4y_3 + ... + 2y_{n-2} + 4y_{n-1} + y_n),$$

- where h = (b-a) / n = 200 ft, n is an even number (8 in our problem), and the y_i (i=0, 1, …) are data elements read from a file.

HW #2 (4)

Here are the main criteria for this problem

- Present the user with a short greeting describing the program.
- Read the name of the data file from the command line in argv[1].
- Open the file, read the data and save it in an array.
- Compute the volume of the lake using Simpson’s rule and the other statistics necessary to find the maximum number of licenses that can be sold.
- Display the final answer with an annotation.

HW #2 (5)

Program requirements

- Use good program style. This includes (but is not limited to the following.

1. Short main function with top-level function calls.
2. Separate interface and implementation files for your globals.
3. Indentation and whitespace.
4. Descriptive names for your functions/constants/variables.
5. Symbolic constants where appropriate.

HW #2 **(6)**

- A namespace around your globals.
- A namespace alias in main to preface your functions invocations.
- Error checking should trap any potential error, such as too few command line args, then print an appropriate message and quit.

// You should time your code, for example, calling

// *gettimeofday()*, and report timing information in

// microseconds.