**Getting Started with Algorithms & Data Structures**

This tutorial will help you get set up on WCIPEG for practice on algorithm and data structure problems. You should already know the basics of a language, like Python or C++ (Python is recommended as your first language). Before working on your first problem, let’s start with a brief introduction.

**Introduction**

Contests like ECOO, USACO, and the CCC are focused on problem-solving using algorithms and data structures. Data structures are methods of storing data (e.g. lists), while algorithms are processes to solve problems using that data (e.g. searching).

These problems are not designed to be language-specific, they focus on solving the problem in any language you want. That being said, Python and C++ are the most commonly used languages; Python is the fastest for you to program in, while C++ runs the program fastest on your computer or the judging server.

WCIPEG is the website we use to practice contest-style problems, where there are thousands of problems available from real and practice contests. When you submit a solution, the online judge will grade it and tell you your score, and you earn points for correct solutions! Be warned: contest problems are extremely strict, your solution must match the problem description **exactly**.

These lessons are designed to help you learn various algorithms and data structures to prepare for contests (like the CCC), assuming you have core programming experience in at least one language.

You can find the lessons on our club resources page under Algorithms & Data Structures. Lessons may contain reading, sample code with explanations, practice problems, and solutions. The main lesson and exercises (usually the first two files) are the most important, but you can always refer to the other examples and solutions.

If you have trouble with these lessons, make sure to ask a friend or club executive for help. And remember, the Internet is your best friend! You should always search Stack Overflow ([stackoverflow.com](http://stackoverflow.com/)) first; it’s a great Q&A website for programmers.

If you need to install or brush up on any languages, see the installation instructions and lessons in the Languages folder on our club resources page.

Let’s get started!

**Make an Account and Log in**

Go to <http://wcipeg.com>. In the top-right corner, click ‘register an account’ and create your account. If you aren’t logged in after registration, click ‘log in’ in the top-right corner and log in.

**Find the Problem**

In the navigation bar at the top of the page, click ‘problems’ and search for ‘a plus b’. Click the problem called ‘A plus B’ (**NOT ‘A plus B 2’**). It should be at or near the top of the list, but the search function is sometimes broken, so check the rest of the list or go to the next page of results if necessary.

You should now be at this page: <http://wcipeg.com/problem/aplusb> (we wanted you to practice searching for problem yourself). Read the problem and try to understand it (don’t worry, these problems can be very technical and confusing). The problem says the following:

“Given two integers a and b, output the sum of a and b. A and B, as well as A+B will be less than 10000 in absolute value.”

This means you need to input two integer numbers (a and b), then output the sum of the two numbers. There will usually be constraints on the numbers, but don’t worry about those for now.

**Solve the Problem**

Now, you need to write the actual program in your editor. Open your program editor, and create a new program (new file in Python or new project in C++). Refer to the Languages folder on the club resources page if you forget how to do that or have any problems.

Here is the Python solution:

a = input()

b = input()

print a + b

Here is the C++ solution:

#include <iostream>

using namespace std;

int main() {

int a;

int b;

cin >> a;

cin >> b;

cout << (a + b) << endl;

return 0;

}

Run the program in your editor to test it (make sure you typed the program correctly and exactly), and enter two numbers (on two separate lines). Your program should output the sum of the two numbers.

**Submit Your Solution**

Assuming it runs correctly on your computer, it’s time to submit it to the judge. Go back to the problem page in your browser. Click Submit on the right side of the page (it should be located below Best Solutions, beside Analysis, and above Clone Problem).

Copy and paste the code from your editor to the code box, or click the Choose File button and select your program file (if you’re using C++, select the main.cpp file). Make sure the Language is set to the language you used (usually Python 2 or C++). Click Submit! After your submission is processed, you should see something like this under Execution Results:

**Test case #1:** AC [0.019s, 5092K] (50/50)

**Test case #2:** AC [0.019s, 5092K] (50/50)

Final score: 100/100

The judge ran the program on 2 different test cases, and both say AC (accepted), so they are both correct! You can see your scores on the different test cases (50/50), as well as your program runtime (0.019s) and memory (RAM) used (5092K = 5092KB = 5.092MB). The final score is 100/100, so you get full points for this problem!

If your program didn’t submit correctly, you can click ‘View source’ to see the code you submitted or ‘Revise and resubmit’ to submit again.

More information on the judge and the possible result status codes: <http://wcipeg.com/wiki/Judge:Help>

**Submission Guidelines**

Some things to keep in mind whenever solving and submitting contest problems:

* Only input and output **exactly what the question says**. A computer is grading your program, so it already knows the exact input and output order specifications and **doesn’t need you to prompt for input** (like outputting “Enter a number:”). If you prompt for input, the judge will consider that as part of your answer output and your entire solution will be wrong.
* Always **test your program** with the given examples before submitting. Your program input and output **must match exactly, character by character**. However, just because your program works for the examples, it may not work for the real test cases.
* If you are using C++ or Java (where you need to create project), only submit the main.cpp or main.java file or code to the grader.

Congratulations! You’ve solved your first contest problem on WCIPEG!