**CYB2200 LAB3-Integer Overflow (Sep 26, 2025)**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(5 pts) Task 1: Run the code to see the length in bytes for each data type.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date type | Normal size | sizeof() | Min value | Max value |
| signed char | 1 byte |  |  |  |
| unsigned char | 1 |  |  |  |
| signed short | 2 |  |  |  |
| unsigned short | 2 |  |  |  |
| signed int | 4 |  |  |  |
| unsigned int | 4 |  |  |  |
| signed long | 4?8? |  |  |  |
| unsigned long | 4?8? |  |  |  |
| signed long long | 8 |  |  |  |
| unsigned long long | 8 |  |  |  |

(5 pts) Task 2: What happens if we assign an out-of-range value to a variable or get an out-of-range value from input?

Your answer:

Task 3: see next page

(15 pts) Task 3: Guess the output for each case first, and then run the code to see the actual output. Document them in the following table.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | Code | Your guess  (output) | Actual output | Overflowed (yes or no) |
| 1a | short x\_short = 20000;  short y\_short = 20000;  cout << "1a. x + y = " << x\_short+y\_short << endl; |  |  |  |
| 1b | x\_short = 20000;  y\_short = 20000;  short z\_short = x\_short + y\_short;  cout << "1b. short z\_short = " << z\_short << endl; |  |  |  |
| 1c | x\_short = 20000;  y\_short = 20000;  int z1\_int = x\_short + y\_short;  cout << "1c. int z1\_int = " << z1\_int << endl; |  |  |  |
| 1d | x\_short = 20000;  y\_short = 20000;  int z2\_int = (int)x\_short + (int)y\_short;  cout << "1d. int z2\_int = " << z2\_int << endl; |  |  |  |
| 2a | int x\_int = 2000000000;  int y\_int = 2000000000;  cout << "2a. x\_int + y\_int = " << x\_int + y\_int ; |  |  |  |
| 2b | x\_int = 2000000000;  y\_int = 2000000000;  int z\_int = x\_int + y\_int; //z is out of range now  cout << "2b. z\_int = " << z\_int << endl; |  |  |  |
| 2c | x\_int = 2000000000;  y\_int = 2000000000;  long long z1\_ll = x\_int + y\_int;  cout << "2c. long long z1\_ll = " << z1\_ll << endl; |  |  |  |
| 2d | x\_int = 2000000000;  y\_int = 2000000000;  long long z2\_ll = (long long)x\_int + (long long)y\_int;  cout << "2d. long long z2\_ll = " << z2\_ll << endl; |  |  |  |
| 3a | long long x\_ll = 2000000000;  long long y\_ll = 2000000000;  cout << "3a. x\_ll + y\_ll = " << x\_ll + y\_ll << endl; |  |  |  |
| 3b | x\_int = 2000000000;  y\_ll = 2000000000;  long long z3\_ll = x\_int + y\_ll;  cout << "3b. long long z3\_ll = " << z3\_ll << endl; |  |  |  |