

Assignment 1 - The Size Of Numbers in C++

Due: Sunday, January 13, 2019, 11:59pm

Number spaces in memory can be declared as unsigned (only holding positive numbers) or signed (holding both positive and negative numbers) in C++. By default, you are given signed number spaces, unless you explicitly declare them as unsigned. In addition, you can specify the exact number of bytes for a number depending on the type used to create the memory spaces, i.e. int, short, and long. This assignment explores the size of different types of numbers to make you aware of your limitations in C++.

Preprocessor Macros (30 pts)

Include the `climits` library to print the preprocessor macros for the max and min sizes of signed and unsigned ints, longs, and shorts. Print the 9 different values for all the types (you don't need to worry about the unsigned min because it is just 0!). Refer to this [cplusplus.com](http://www.cplusplus.com/reference/climits/?kw=climits) reference (read carefully!!!): <http://www.cplusplus.com/reference/climits/?kw=climits>

Using Functions and Calculations: `sizeof()` and `pow()` (50 pts)

Suppose you didn't have macros, but you had a function that told you the number of bytes for a type. Print the calculated values for the largest and smallest signed short, int, and long, and the largest unsigned short, int, and long based on calculations using the number of bytes.

The `sizeof()` function returns the number of bytes used by the type supplied to the function. For example, you can use this function by passing the type as an argument, i.e. `sizeof(short)`, `sizeof(int)`, etc. Now, you know how to calculate how many bits are in a byte!

In order to calculate an exponent, we need to include the `cmath` library (<http://www.cplusplus.com/reference/cmath/>) and use the `pow()` function (<http://www.cplusplus.com/reference/cmath/pow/>). Example is shown below:

```
#include <cmath>
#include <iostream>
int main() {
    std::cout << "Pow 2, 8: " << pow(2, 8) << std::endl;
    return 0;
}
```

Make sure you label your output and that these numbers match the numbers printed from the preprocessor macros!!!

(10 pts) Program Style/Comments

In your implementation, make sure that you include a program header (5 pts) in your program, in addition to proper indentation/spacing and other comments (5 pts)! 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/winter2019/cs161-001/assignments/161_style_guideline.pdf

```
/******  
** Program: numbers.cpp  
** Author: Your Name  
** Date: 01/13/2019  
** Description:  
** Input:  
** Output:  
*****/
```

(10 pts) **Predictive Pre-Peer Review Survey:**
http://oregonstate.qualtrics.com/jfe/form/SV_a5nstixZt2F5Lpj

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

<https://go.peerceptiv.com>

Class code is **happen12**

*****If you are doing this off campus, pay attention to the off-campus directions from lab!**