# ECE 263

**Lab 2**

**Due: Thursday, September 5th, 2013 at 2:00pm**

## Warm-up Exercise 1: Data type

## 1.Fill in the table below ( Note: No need to memorize but try to get familiar with them)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Datatypes in C | | | | | | | |
| signed/unsigned | Datatypes | Size(Bytes) | Size(Bits) | Range | Specifier | Octal | Hex |
| signed | char |  |  |  |  |  |  |
| unsigned |  |  |  |  |  |  |
| signed | short |  |  |  |  |  |  |
| unsigned |  |  |  |  |  |  |
| signed | int |  |  |  |  |  |  |
| unsigned |  |  |  |  |  |  |
| signed | long |  |  |  |  |  |  |
| unsigned |  |  |  |  |  |  |
|  | floatdouble |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

2.Display the minimum value for int and the maximum value for a float data-type

## 

## Warm-up Exercise 2: Flag Specifiers

Write a program that will read an integer and a floating-point value from the keyboard with

a single scanf function.

Follow the lab instructors and output the data in a formatted manner with flag specifies

## Warm-up Exercise 3: Characters and Ascii Table

Write a program that will read information from the keyboard

1. A character and store it in a variable that is data-type char
2. Output the values and display the ASCII code for the character that was entered

## Warm-up Exercise 4: Constant

## Follow the lab instructor instructions to define constants in two ways.

## Warm-up Exercise 5: printf() scanf()

1. Write a program that will show a message (any message you like) to the screen, then output how many characters were displyed by the printf() function
2. Write a program that will read an integer (a three-digit integer in the form of **abc**) from the keyboard and display each single digit. Then output the sum of the three digits.

* Use integer division and modulus
* Try scanf() to read one character at a time

## Warm-up Exercise 6: Operators

Follow the lab instructor instructions to do the following exercises:

1. Arithmetic operations and precedence: Calculate the value of z

z = 12\*6+20\*5

z=12/3\*2

z=12/(3\*2)

1. Increment and decrement operators: Calculate the value of a

x=7, y=5, z=3, a = y + ++x + z;

x=5, y=1, z=3, a=--y+x+z;

x=5,y=1,z=3,a=y--+x+z;

Try the code below, do they work as expected?

* int a = 0;

printf ("%d %d", a++, ++a);

* int a = 0, b;

b = ++a + ++a; printf ("%d", b);

1. Abbreviated assignment:

x=7, x - = 3;

x=3, y=2, x += y;

x=5, y=2 z=3 x+= y + ++z;

x=5, y=1 z=3 z -= y + x;

x=4, y=2 z=3 x \*= y + z;

1. Integer division and type-casting

Try the code blow:

int sum;

int num1 = 5.6;

int num2 = 5.8;

sum = num1 + num2;

printf(“%d\n”,sum);

sum=(int)num1 + (int)num2;

printf(“%d\n”,sum);

# Lab2 Assignment

## For each of the programs assigned below submit the following:

* + Hard copy of the source code (.c file)
  + Place the .c file in Oncourse under Assignments. (Do not submit the entire project JUST the .c file)

## Guidelines:

* + The .c files should be named in the following format:
    - LastName\_LabNumber\_ProblemNumber.c

e.g. Ling\_Lab2\_Problem1.c

* + Begin each program with a comment section that contains the following:
    - ECE263 or ECE264
    - Lab Number: Lab 2
    - Problem Number: Problem 1
    - Author’s Name: Jie Ling
    - Description:
    - A description of what the problem does
  + Use comments throughout the source code to explain the steps.
  + In main indent statements to be executed
  + In general, use lower case letters for variable names. Variable names should be mnemonic. Separate words in a variable name with an underscore or begin the second word with an uppercase letter.
  + Place no more than one C statement per line.
  + Do not type more than 77 characters per line.

## Problem 1:

You are to program a billing software for AT&T “Pay as You Go” customers. All the services are to be calculated basing on the usage of services.

Services from AT&T Wireless include:

Domestic calls (minutes)

International calls(minutes)

Data (MB)

Text Messages

Voice Messages

The unit prices are $0.19, $1.00, $1.00, $0.20, $0.25 respectively.

Assume an agent is going to enter information manually via keyboard. You program will support

the follows:

1. Read the amount of usage for each service from the keyboard.
2. Calculates the cost of each service, the subtotal, and the total cost after an **8.25%** sales tax.
3. Add **$5** activation fee;
4. Subtract **$10** promotion
5. Output the result in the form below.

|  |  |  |  |
| --- | --- | --- | --- |
| QTY | DESCRIPTION | UNIT PRICE | TOTAL PRICE |
| ----- | --------------------- | ----------------- | -------------------- |
| Xxx | New Prepaid Phone | 200.00 | xxxxx.xx |
| Xxx | Domestic calls (minutes) | 0.19 | xxxxx.xx |
| Xxx | International calls(minutes) | 1.00 | xxxxx.xx |
| Xxx | Data (MB) | 1.00 | xxxxx.xx |
| Xxx | Text Messages | 0.20 | xxxxx.xx |
| Xxx | Voice Messages | 0.25 | xxxxx.xx |
|  |  |  | -------------------- |
|  |  | LABOR DAY PROMOTION pRPROMOTION PROMOTO | -xx.xx |
|  |  | ACTIVATION FEE | +xx.xx |
|  |  | SUBTOTAL | +xxxxx.xx |
|  |  | TAX | +xxx.xx |
|  |  | TOTAL | +xxxxx.xx |

* Use a manifest (symbolic) constant for the tax rate and constant modifier (const) for the unit prices.
* Use data-type unsigned short to store the quantities.
* The output should be aligned in the following format:

(Don’t use Spaces but flags)

* + Output the column “qty” with leading zeros
  + Output the column “description” with spaces on the RHS
  + Output the column “unit price” with leading blank spaces;
  + Output the value of “Activation fee” “total price” “subtotal” “tax” “total” with leading spaces.
  + Display the + - sign for the value of “Activation fee” “total price” “subtotal” “tax” “total”