Lab 3

You are expected to copy and paste your code into each corresponding box in this handout and submit it as a Word document or PDF file before the due. Additionally, your lab instructor will tell you which three questions you must showcase during the lab session. While you may demonstrate your code running in person after the due date, your file must be submitted on time.

Hint #1: For questions asking for 'a function which takes in xyz', the xyz refer to function parameters, not the use of scanf().

Hint #2: For questions asking for 'a program', you are expected to include all necessary #include and define your main function. It's up to you whether to define additional helper functions, though usually, you don't need to.

Task 1: Write a **function** which takes in an integer, and then prints the triple of its value (Eg, if input is 2, prints 6)

```
void print_triple(int number) {
    printf("%d\n", number * 3);
}
```

Task 2: Write a **function** which takes in two integers, and returns the smaller one

```
int min(int number_1, int number_2) {
    return (number_1 < number_2) ? number_1 : number_2;
}
```

Task 3: Write a **function** which takes in a float, prints it, and then return the absolute value of it.

```
float print_abs(float number) {
    float abs = (number < 0) ? -number : number;
    printf("%f\n", abs);
    return abs;
}
```

Task 4: Write a **function** which takes in a char, and prints the input char 100 times

```
void print_100_times(char c) {
    for (size_t i = 0; i < 100; ++i) {
        printf("%c\n", c);
    }
}
```

Task 5: Write a **function** which takes in two doubles, and prints the non-negative difference (Eg, the bigger one – the smaller one) [Hint: if the two inputs are the same, prints 0]

```
void print_difference(double number_1, double number_2) {
    double difference = number_1 - number_2;
    printf("%f\n", (difference < 0) ? -difference :
            difference);
}
```

Task 6: define a function which takes in two integers, and then prints the result of the first number got divided by the second one as a double (Eg, if input 1 4, should print 0.25)

```
void print_divide(int dividend, int divisor) {
    printf("%f\n", (double) dividend / divisor);
}
```

Task 7: define a **program** which reads in a double, and then prints it out as three times: (1) As an int (2) as a double (3) as a float

```
#include <stdio.h>

int main() {
    double input;
    printf("enter a double:\n");
    scanf("%lf", &input);

    printf("%d\n", (int) input);
    printf("%f\n", input);
    printf("%f\n", (float) input);

    return 0;
}
```

You are required to use function from math library to answer Task 8 ~ 10:

Task 8: define a **program** which reads in two doubles, and then prints out the one which has bigger absolute value

```c
#include <stdio.h>
#include <math.h>

int main() {
    double first;
    double second;
    printf("enter first double:\n");
    scanf("%lf", &first);

    printf("enter second double:\n");
    scanf("%lf", &second);
    first = fabs(first);
    second = fabs(second);

    printf("%f\n", (first < second) ? second : first);

    return 0;
}
```

Task 9: define a **program** which reads in one positive float (called x), and then prints the value of sin(2x) [Hint: sin(2x) = 2 sin(x) *cos(x)]

```c
#include <stdio.h>
#include <math.h>

int main() {
    float x;
    printf("Enter a positive float: \n");
    scanf("%f", &x);

    printf("%f\n", sin(2*x));

    return 0;
}
```

Task 10: define a **program** which reads in two positive integers, and then prints their log (base 10) of their sum [Assumption: the user will type in two positive integers]

```c
#include <stdio.h>
#include <math.h>

int main() {
    int first;
    int second;
    printf("enter first positive integer:\n");
    scanf("%d", &first);

    printf("enter second positive integer:\n");
    scanf("%d", &second);

    printf("%f\n", log10(first + second));

    return 0;
}
```

Task 11: define a **program** to randomly generate and then print a number in range between 4 to 9

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main() {
    srand(time(NULL));
    int num = (rand() % ((9 - 4) + 1)) + 4;
    printf("%d\n", num);

    return 0;
}
```

Task 12: define a **program** to randomly generate and print the top value on a six-sided die (1 ~ 6) for 20 times (Hint: top value is the value on the side facing up)

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main() {
    srand(time(NULL));

    for (size_t i = 0; i < 20; ++i) {
    int num = (rand() % 6) + 1;
    printf("%d\n", num);
    }

    return 0;
}
```

Task 13: define a **program** to calculate 10! and print the result, using recursion [hint: int may not be big enough, you may want to use long]

```
#include <stdio.h>

// calculate n!
long int factorial(long int n) {
    return (n == 1) ? n : n * factorial(n - 1);
}

int main() {
    printf("%ld\n", factorial(3));
    return 0;
}
```

COMP 2510

Task 14: define a **program** which reads in one positive integer from user, and then print out the input number in reversed order, using recursion
(Eg, when input is 123, output should be 321; when input is 100, output can be 001 or just 1)

```c
#include <stdio.h>

int print_reverse(int number) {
    if (number < 10) {
    return number;
    } else {
    printf("%d", number % 10);
    return print_reverse(number / 10);
    }
}

int main() {
    printf("%d\n", print_reverse(100));
    return 0;
}
```

Task 15: define a **program**, using recursion, to calculate and print the first 20 numbers in Fibonacci sequence [hint: 0 1 1 2 3 5 8 13 21 34 … (since the third number, each one is the sum of the previous two)]

```c
#include <stdio.h>

// calculate the nth fibonacci number.
// 1-indexed.
int fibonacci(int number) {
    if (number == 1) {
        return 0;
    } else if (number < 4) {
        return 1;
    } else {
        return fibonacci(number - 1) + fibonacci(number -
                                                2);
    }
}

// print the fibonacci numbers from start to end
void print_fibonnaci(int start, int end) {
    if (start == end) {
        printf("%d\n", fibonacci(start));
    } else if (start < end) {
        printf("%d ", fibonacci(start));
        print_fibonnaci(++start, end);
    }
}

int main() {
    print_fibonnaci(1, 20);
    return 0;
}
```