Programming Design Worksheet - Redfield

for CS1310 (programs 2-7) and CS1311 (programs 1-6)

Copy this file. Type and past images to create new documents for designs.
Print it for class (if you must miss, submit one file to Designs).

First name **Davide**   Last name **Russillo**

Design for program name **Factoring**

DATA

Variables needed in **WORDS for main** and **globally**

first input
second input
restart?

Formulas/equations + if any

C ***DECLARATIONS*** for main & global

```
int input_a;
int input_b;
int restart = 0;
```

(STARTING TicTacToe:put image; or draw: Insert, Drawing; or put at end of the file)

draw in RAM with possible values

Algorithm to PSEUDOCODE level for each function

(*remember to indent under* if, switch, while, do-while, for)

main:

```
do
```

```
    input_a = get_positive_integer()
    input_b = get_positive_integer()
    print your inputs are ‘input_a’ and ‘input_b’!
    print the factors for a are
    get_factors(input_a)
    print the factors for b are
    get_factors(input_ b)
    print the prime factors for a are
    get_prime_factors(input_ a)
    print the prime factors for b are
    get_prime_factors(input_ b)
    print the lcm is
    get_lcm(input_a, input_b)
    print the gcd is
    get_gcd(input_a, input_b)
    print enter 0 to exit or enter 1 to restart
    scanf -> restart
while restart is equal to 1
```

`other functions (`**`bold the names`**`): (put them before main in the program!)`

```
int get_positive_integer(void)
    int input = 0;
    do
        print input the value:
        scan -> input
        print your input is ‘input’
        if input not positive
            print invalid input!
    while input not positive
    return input

void get_factors(int value)
    int i
    for i = 2; i < value; increment i
        if value divides into i
            print i
    print value
```

```
void get_prime_factors(int num)  /* from dr. Redfield */
    initialize count = 1
    while count < num
        increment count
        if count divides into num
            print count
            num = num / count
            decrement count

void get_lcm(int value_a, int value_b)
    int base
    int lcm
    if value_a > value_b
        base = value_a
    else
        base = value_b
    lcm = base
    while lcm is not divisible by value_a or value_b
        increment lcm by base
    print the lcm is ‘lcm’

void get_gcd(int value_a, int value_b)
    int gcd = 0
    int i
    for i = 1; i <= value_a and i <= value_b; increment i
        if value_a and value_b are divisible by i and i > gcd
            gcd = i
    print gcd
```