ENGR 101 – Introduction to Programming
Spring 2019
Study Questions – Week 6

Q1 - Fix the error in the following code.

```
def square(x):
    result = x*x
    print result
print square(5)*square(5)
```

Q2. Write a function which takes n digits number as an input and return True if the biggest digit in this number is divisible by 3.

Q3 - Write two functions *quotient ()* and *remainder ()* that take two arguments each, referred to as numerator and denominator. When we call the functions, these should be the ouputs:

quotient(10,3)

3

remainder(10,3)

1

Now write another function that takes two inputs which are again numerator and denominator. In this function if the remainder is 0 then this function should return the result of the *quotient ()* function. Else if the remainder is not zero, the function should print the message "The number cannot be divided without remainder". (Inside this function use *quotient() and remainder()* functions)

Q4 - Write a function *fahrenheit()* that takes a temperature value in celsius and converts it to a fahrenheit value. The formula is **T_in_fahrenheit = (T_in_celsius * 9 / 5) + 32.** Then write another function *fahrenheit_range()* that finds the fahrenheit values of celsius degrees from 22 to 27 degrees, by using the *fahrenheit()* function. The function should print the following.

22.0: 71.6

23.0: 73.4

24.0: 75.2

25.0: 77.0

26.0: 78.8

27.0: 80.6

Q5 - Write a function which finds the factorial of the given number as an argument. Use while loop.

Q6 - Write a function *area()* that takes radius as an input and finds the area of a circle. The formula of finding area of circle is A = pi * radius**2. Then write another function *distance_and_area()* that takes the coordinates of two points x1, y1, x2, y2 as an input. First point is the center and second point is the perimeter of the circle. The function computes the area of this circle by using *area()* function. (Use **math** library for pi and for taking the square root)

for example;

print distance_and_area(8,7,4,4)

78.53975

Q7 – 7. Write a function add_even_numbers() that takes two integers as arguments and prints the sum of even numbers between them.

Hint: x%2==0 if x is even

add_even_numbers(5,12)

>> 24

add_even_numbers(4,7)

>>6

Q8 - Create a function **greedy_prime** that takes an integer n and checks if the number is prime. A prime number has only two factors: 1 and the number itself. The example function calls are produced below.

>> greedy_prime(10)

10 is not a prime number

>> greedy_prime(919)

919 is a prime number

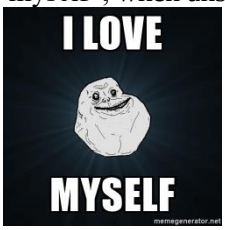
Q9 - Create a function, it takes 3 integer inputs (number1, number2, number3). This function is going to check all number s from 0 till number1, if they are divisible by number2 and number3.

Q10 - Figure out the error or errors in the below code and after fixing errors what will be the output?

```
age = raw_input(Please enter your age:)
if age = 18:
        print “you are able to vote in the United States!
else:
        print “You are not able to vote in the United States!
```

ENGR 101 – Introduction to Programming
Spring 2019
Study Questions – Week 6

Q11 - Write a function, which asks user who the user loves. If the answer of the user is "myself" it prints "I love myself". Otherwise it prints "Wrong answer!". It should ask it till the answer is "myself", when answer is "myself" it stops asking.



Q12- Write a function that will ask the user to enter his height and returns True if the user's height is an even number and False if it is an odd number.

Q13 – Try to edit this infinity loop in 2 ways: You code should prints 4 only one time in both ways.

```
while True:
    print(4)
```

Q14 - Write a *guess()* function which does the following steps;

a) It chooses a random number between 0 and 100 (Hint: There is a module called random in Python. Import it, and use the function randint(a, b) from this module)

b) asks for user input

c) checks whether the number that is entered matches with the generated number

d) If so, terminate the program

e) If not, tells whether the input is smaller or larger than the generated one and continues asking for a new guess, unless the generated number is guessed.

Q15 – Our solution for the previous question(Q14) is too long ! E.g. we wrote the following line three times:

```
input = int(raw_input("Enter a number:"))
```

Can we write it only one time and get the same output?

Q16 - Write a function *change_username()* that helps you to change your username in a website.

Your program should work as shown below:

First, it should ask you to enter your current username:

>> Enter your current username:

If the username is true, it should ask you to enter your password. If your user name is "KRAL" you should prompt as follows:

>> Enter your password for the KRAL:

Then it should prompt you to enter your new username

>> Enter your new username:

And as soon as you enter your new username, your program should terminate.

Q17 - What is the output of the following code?

```
def func(a, b, c):
    if a == b:
        return a
    elif b == c:
        return b
    elif a == c:
        return c
print(func(3, 5, 5))
```

Q18 - Write a Python program that takes a number n as input and produces a half-pyramid-like output as shown below. Note that the following output is given when the input to the function is 5. Your function should produce a similar output for any positive number n.

0

01

012

0123

01234

Q19 – In the following code, the user enters a number then we check if the number is equal or not. Still, if we enter 3 in the console, the code below the first if statement will not be executed, and the program will print False. Why? Can you fix this problem?

```
num = raw_input("Enter a number:")
if num == 3:
    print True
elif num != 3:
    print False
```

Q20 – What is the output of the following piece of code?

```
a = 10
b = 5

if a > b:
    c = 3
while a < b:
    c +=1
    if a > b:
        a = 7
        b = 6
        break

while a > 2:
    c += 2
    a -=5
    while c > 1:
        c -=2
        a +=1
        b -=1
        break
print a
print b
print c
```