**ICS3U-Advanced Programming Assignment-Functions**

**\_\_\_\_ out of 45 (TI)**

**Instructions:** Working in your groups complete the questions in this handout. All your answers must be in red font. YOU CAN NOT USE THE INTERNET AS A RESOURCE AND YOU CAN NOT WRITE AND TEST ANY CODE. When you are done, upload this file. Make sure to first save it using the filename format: **yourNamesFA.docx**. 1 mark will be deducted per hour after 4 pm for late submissions.

**Section A**: Functions have been designed. Function calls will be made. Your job is to determine the value returned by each function. Write your answers in the adjacent boxes. Two marks each (if you are wrong, you may be awarded 1 mark for showing your work).

**Problem 1:**

Return 128

function generate(x, y)

{

let a=x\*\*y;

return a;

}

***generate(2,7);***

**Problem 2:**

Return 50

function generate(x, y)

{

return (+x++y);

}

***generate(20,30);***

**Problem 3:**

function generate(x, y)

{

if (+x>+y)

return (true);

else

return false;

}

***generate(15,9);***

**Problem 4:**

function generate( x, y)

{

return (+x+ “ “ + +y);

}

***generate(12,23);***

**Problem 5:**

function generate(x)

{

let sum=1;

for(let c=0;c<x;c++)

{

sum\*=x;

}

return (sum);

}

***generate(4);***

**Problem 6:**

function generate(x, y, a)

{

if(a>=x && a<=y)

{

return (true);

}

else

{

return (false);

}

}

***generate(1,2,1);***

**Problem 7:**

function generate(n)

{

return (Math.round(100+Math.random()\*n)>0);

}

***generate(150);***

**Problem 8:**

function generate(name,count)

{

let result=””;

for(let x=0;x<4;x++)

{

result+=”Lisa” + ”-“ + “Lisa” + ”\n”;

}

Lisa-Lisa

Lisa-Lisa

Lisa-Lisa

Lisa-Lisa

Lisa-a

\

return result;

}

function generate(name,count)

{

let result=””;

for(let x=0;x<count;x++)

{

result+=name +”-“+name+”\n”;

}

return result;

}

***generate(“Lisa”,4);***

**Problem 9:**

function generate(character, y, x)

{

let output=””;

for(let c=0; c<5; c++)

{

for(let n=0; n<2; n++)

{

output +=”x”

}

output +=”\n”;

}

return (output);

}

***generate(“x”,5,2);***

xx

xx

xx

xx

xx

function generate(character, y, x)

{

let output=””;

for(let c=0; c<y; c++)

{

for(let n=0; n<x; n++)

{

output +=character

}

output +=”\n”;

}

return (output);

}

***generate(“x”,5,2);***

**Problem 10:**

return false

function generate(character)

{

if(character<”h”)

{

return (true);

}

else

{

return (false);

}

}

***generate(“z”);***

**Section B:** Write functions for each of the following (example calls to each function are provided). Marks will be deducted for poor programming style i.e. naming, formatting, spacing etc. Documentation is not required. 5 marks each.

**min(10,9,11)🡪**takes 3 numbers and returns the smallest.

function min(num1,num2,num3)

{

if (num1<num2&&num1<num3)

{

return num1;

}

else if (num2<num1&&num2<num3)

{

return num2;

}

else

{

return num3;

}

}

**average(4,5,6)🡪**takes 3 numbers and returns its average.

function average(num1,num2,num3)

{

let average=(num1+num2+num3)/3;

return average

}

**invest(4590,10.7,5)🡪**takes an investment amount, an interest rate as a percentage and a term in years and returns how much the investment will be worth at the end of the term. Interest is compounded annually. As an example if someone invests $100.00 at 10% for 5 years, they would have the following(i.e. the function will return 161.05):

year 1 $110

year 2 $121

year 3 $133.1

year 4 $146.1

year 5 $161.05

function invest(investAmount,rate,years)

{  
 let rate\*=0.01;

let counter=0;

let total=investAmout;

while(counter<years)

{

total+=total\*rate;

counter++;

}

return total;

}

**matrix(10)🡪**takes a single number and returns a matrix of random 1’s and 0’s that has width and length equal to the argument. Here’s an example:

matrix(10) would return the following:

1 0 0 1 1 0 0 0 1 1

0 0 1 0 1 0 1 0 0 0

0 1 0 1 0 0 0 0 0 1

1 1 1 0 0 0 0 1 1 1

1 1 0 1 1 1 0 1 0 0

1 0 0 0 1 1 0 0 0 0

0 0 1 0 0 0 0 1 1 1

1 1 0 1 0 1 0 0 1 0

0 0 1 0 0 0 0 1 1 0

1 1 1 0 0 1 1 1 1 0

**prime(5)🡪**takes a single number and returns true if it is prime or false if it is not