Assignment 1 – Questions

Q1.

A) False, printf doesn't start a new line without using escape sequence "\n"
B) True
C) True
D) True
E) False, variables are case-sensitive
F) True
G) True
H) False, "*", "/", and "%" have higher precedence than "+" or "-"
I) False, "\n" can be used to create new lines within on printf
J) False, only arguments after scanf require an ampersand

Q2.

A) Missing "%" before conversion specification and "&" before variable.

   Corrected: scanf("%d", &value);

B) "\n" outside of quotations, not enough output items( needs 3, only has 2)

   Corrected: printf("The product of %d and %d is %d\n", x , y ,z);

C) Statements are arranged wrong, variable should be on left, equation on right. Should also have semi-colon

   Corrected: sumOfIntegers = firstNumber + secondNumber;

D) Multi-line comment symbols are the wrong way around

   Corrected: /*Program to determine the largest of three integers*/

E) Scanf has a capital S, variable is missing ampersand

   Corrected: scanf("%d", &anInteger);

F) Variable and conversion specification amount don't match, last variable shouldn't have ampersand, missing comma between last 2 variables

   Corrected: printf("Remainder of %d divided by %d is\n", x, y);

G) Printf is missing an f, comma is on the wrong side of the quotations

   Corrected: printf("The sum is %d\n", x+y);

H) Printf has capital p, missing end quotation, variable has unnecessary ampersand

   Corrected: printf("The value you entered is: %d\n", value);

I) Variable doesn't need ampersand

   Corrected: printf("The value is %d\n", number);

J) Second variable is missing ampersand

Corrected: scanf(“%d%d”, &number1, &number2);

Q3.

A) int c, thisVariable, q76354, number;
B) printf(“Enter an integer: “);
C) scanf(“%d”, &a);
D) printf(“This is a C program”);
E) printf(“This is a C\nprogram”);
F) printf(“This\nis\na\nC\nprogram”);
G) printf(“This\tis\ta\tC\tprogram”);
H) a = b * c;
I) //Purpose: Performs sample payroll calculation
J) scanf(“%d%d%d”, &a, &b, &c);

Q4.

A, D, and E are correct. C’s logic regarding binary operators makes these 3 equivalent.

Q5.

A) x = 7 + 3 * 6/2 -1;

First comes multiplication (highest precedence and left associative)

x = 7 + (3*6)/2 – 1;

Next comes division(also highest precedence but was not leftmost operation)

x = 7 + ((3*6)/2) – 1;

Next comes addition, followed by subtraction (same precedence but left associative)

x = (7 + ((3*6)/2)) – 1; x = 15

B) x = 2 % 2 + 2 * 2 – 2 / 2;

First modulus, then multiplication, and then division, in that order (equal precedence, left associative)

x = (2 % 2) + (2 * 2) – (2/2);

Then addition followed by subtraction as they are also left associative

x = ((2 % 2) + (2 * 2)) – (2/2); x = 3

C) x = (3 * 9 * (3 + (9 * 3 / (3))));

This equation has already been organised, so it just follows regular convention, working from the center outwards. First is 3/3

x = (3 * 9 * (3 + (9 * 1)))

Next comes 9 by 1

x = (3 * 9 * (3 + 9))

Next is 3 + 9

x = (3 * 9 * (12))

Finally, 3 by 9 by 12

x = 324