QUIZ 2 RUBRIC

Q1. One of the possible solution code(s) is the following:

```
1. mid_fun:                    //defining the function name
2. mov r5 $2                   //initializing r5 = 2
3. mov r3 $0                   //initializing r3 = 0
4. mov r4 $1                   //initializing r4 = 1
5. add r8 r3 r2                //copying the number to be divided in r8; redundant
6. sub_again:
7. sub r15 r15 r5              //deducting 2 from dividend, following long division
8. add r14 r14 r4              //incrementing the quotient by 1
9. beq r15 r4 r7               //will branch to r7 if remainder of r15 is 1
10.beq r15 r3 r7               //will branch to r7 if remainder of r15 is 0
11.b sub_again                 //unconditional branch
```

Q2.

A. One possible Pseudocode is:

main:
  int **n** ;
  input taken in variable **n**;
  int **m** = ***mid_fun***(**n**);
  for (i=0;i<**m**;i++)
        {
                for(j=1 ; j<=(**i+1**) ; j++)
                {
                        print (" * ");
                }
                print ("**\n**");
        }
  // now i is = 2
  for (k=**i**;k>0;k--)
        {
                for(p=1 ; p<=(**k+1**) ; p++)
                {
                        print (" * ");
                }
                print ("**\n**");
        }

B. The final output of the assembly code on execution is:
```
*
**
***
**
```

Q3.

		F	D	E			
1. mov r0, r2		F	D	E			
2. mov r1, r3			F	D	E		
3. add r4, r8,r9				F	D	E	
4. add r5, r2, r3					F	D	E

(1.25*12 = 15)

Q4.   (1.5*10 = 15)

	Hex	**Decimal**
1. PUSH r1	0x3FF	1023
2. PUSH r2	0x3FE	1022
3. POP r1	0x3FF	1023
4. PUSH r1	0x3FE	1022
5. PUSH r5	0x3FD	1021
6. PUSH r4	0x3FC	1020
7. POP r2	0x3FD	1021
8. PUSH r3	0x3FC	1020
9. POP r5	0x3FD	1021
10. PUSH r7	0x3FC	1020