Problem 1:

Convert the following c-code to MIPS assembly code.

```
int x =10;
int y =5;
int  z=0;
if (x>y)
        z = x-y;
else
        z = x+y;
```

Problem 2:

Convert the following c-code to MIPS assembly code.

```
int a =5;
int b =6;
int max =0;
if (a<b)
        max = b;
else if (a>b)
        max = a;
else
        max =100;
```

Problem 3:

Write a MIPS Assembly program that finds the maximum value within 3 values, given the following c-code.

```
int a =5;
int b =6;
int c =8;
int max =0;
if (a>b)
        if (a>c)
                max =a;
        else
                max = c;
else
        if (b>c)
                max = b;
        else
                max =c;
```

Problem 4:

Write a MIPS assembly program that is equivalent to the given high level programming language code below.

```
int sum =0;
int avg =0;
for (int i=1; i<=20; i++)
{
        sum +=i;
}
avg = sum/20;
```

Problem 5:

Write a MIPS assembly program that calculates the value of 5^3, given the high-level programming language code below.

```
int a =1;
for (int i=0; i<3; i++)
{
        a*=5;
}
```

Problem 6:

Write a MIPS assembly program that sums up the numbers from 1to 10 starting from number 10, given the high-level programming language code below.

```
int sum =0;
for (int i=10; i>0; i--)
{
        sum +=i;
}
```

Problem 7:

Write a MIPS assembly program to find the greatest common divisor (GCD) of 2 integers, where the GCD is the largest positive integer that divides the numbers without a remainder. For example, the GCD of 8 and 12 is 4, the GCD of 54 and24 is 6. The below high level programming code is a simple way to find the greatest common divisor (GCD) of a and b.

```
int a = 54;
int b =24;
while(a != b)
{
        if(a>b)
          a=a-b;
        else
            b=b-a;
        }
```