

**Task 01:** Submit a comprehensive commented file of the original code

```
#include <stdint.h> //variable definitions for the C99
#include <stdbool.h> //boolean definitions for the C99
#include <math.h> //code uses the sinf() function prototyped
#include "inc/hw_memmap.h" //macros defining the memory map
#include "inc/hw_types.h" //defines common types and macros
#include "driverlib/fpu.h" //support for Floating Point Unit
#include "driverlib/sysctl.h" //defines and macros for System Control API

#include "driverlib/rom.h" //

#ifndef M_PI
#define M_PI 3.14159265358979323846
#endif

#define SERIES_LENGTH 100 //the depth of data buffer

float gSeriesData[SERIES_LENGTH]; //an array of loads SERIES_LENGTH long

int32_t i32DataCount = 0; //a counter for our computation loop

int main(void)
{
    float fRadians; //float variable to calculate sine

    FPU_LazyStackingEnable(); //turn on Lazy Stacking
    FPU_Enable(); //turn on FPU
    //set up system clock for 50MHz
    SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
SYSCTL_OSC_MAIN);

    fRadians = ((2 * M_PI) / SERIES_LENGTH); //divide 2pi by the depth of array

    //while loop to calculate the sine value for each of the 100 values of the angle
    while(i32DataCount < SERIES_LENGTH)
    {
        gSeriesData[i32DataCount] = sinf(fRadians * i32DataCount);

        i32DataCount++;
    }
    //endless loop
    while(1)
    {
    }
}
```

**Task 02:** Modify the code to implement the below equation with 1000 sample points and determine the time for FPU operation.

```
#include <stdint.h> //variable definitions for the C99
#include <stdbool.h> //boolean definitions for the C99
#include <math.h> //code uses the sinf() function prototyped
#include "inc/hw_memmap.h" //macros defining the memory map
```

```
#include "inc/hw_types.h" //defines common types and macros
#include "driverlib/fpu.h" //support for Floating Point Unit
#include "driverlib/sysctl.h" //defines and macros for System Control API

#include "driverlib/rom.h" //

#ifndef M_PI
#define M_PI 3.14159265358979323846
#endif

#define SERIES_LENGTH 1000 //the depth of data buffer to hold values is 1000

float gSeriesData[SERIES_LENGTH]; //an array of loads SERIES_LENGTH long

int32_t i32DataCount = 0; //a counter for our computation loop

int main(void)
{
    FPU_LazyStackingEnable(); //turn on Lazy Stacking
    FPU_Enable(); //turn on FPU
    //set up system clock for 50MHz
    SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
SYSCTL_OSC_MAIN);

    //while loop to calculate the sine value for each of the 100 values of the angle
    while(i32DataCount < SERIES_LENGTH)
    {
        gSeriesData[i32DataCount] = (-0.06366) +
(0.12613*(cosf(M_PI*i32DataCount/500)))+(0.12258*(cosf(M_PI*i32DataCount/250)))
+(0.01593*(sinf(M_PI*i32DataCount/500)))+(0.03147*(sinf(M_PI*i32DataCount/250)));

        i32DataCount++;
    }
    //endless loop
    while(1)
    {
    }
}
```

Identity	Name	Condition	Count	Action
<input checked="" type="checkbox"/> Count Event	Count Event		39250	
<input checked="" type="checkbox"/> main.c, line 33 (\$C\$) Breakpoint			0 (0)	Remain Halted
<input checked="" type="checkbox"/> main.c, line 39 (\$C\$) Breakpoint			0 (0)	Remain Halted