Steven Lee
Github root directory: https://github.com/lees106/STAT411

Date Submitted: 10/21/2019


Lab9_Video_Links.txt

Task 01:

Youtube Link:
https://youtu.be/1jGaAS1A9ck

Modified Code:

```
#include <stdint.h>
#include <stdbool.h>
#include <math.h>
#include "\ti\tivaware_c_series_2_1_4_178\inc\hw_memmap.h"
#include "\ti\tivaware_c_series_2_1_4_178\inc\hw_types.h"
#include "\ti\tivaware_c_series_2_1_4_178\driverlib\sysctl.h"
#include "\ti\tivaware_c_series_2_1_4_178\driverlib\rom.h"
#include "\ti\tivaware_c_series_2_1_4_178\driverlib\fpu.h" //Supports floating point

#ifndef M_PI
#define M_PI 3.14159265358979323846
#endif

#define SERIES_LENGTH 100 //depth of buffer

float gSeriesData[SERIES_LENGTH];
int32_t i32DataCount = 0; //Counter for computation loop

int main(void)
{
	float fRadians; //Used to calculate sine
	ROM_FPULazyStackingEnable();
	ROM_FPUEnable(); //From reset this is off
	ROM_SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
	SYSCTL_OSC_MAIN); //50Mhz
	fRadians = ((2 * M_PI) / SERIES_LENGTH);
	while (i32DataCount < SERIES_LENGTH)
	{
		gSeriesData[i32DataCount] = sinf(fRadians * i32DataCount); //sinf() from
		math.h
		i32DataCount++;
	}
	while (1)
	{
	}
}
```

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.

Steven Lee
Github root directory: https://github.com/lees106/STAT411

Task 02:

Youtube Link:
https://youtu.be/NRe3mv1Pq-E

Modified Code:

```
#include <stdint.h>
#include <stdbool.h>
#include <math.h>
#include "\ti\tivaware_c_series_2_1_4_178\inc\hw_memmap.h"
#include "\ti\tivaware_c_series_2_1_4_178\inc\hw_types.h"
#include "\ti\tivaware_c_series_2_1_4_178\driverlib\sysctl.h"
#include "\ti\tivaware_c_series_2_1_4_178\driverlib\rom.h"
#include "\ti\tivaware_c_series_2_1_4_178\driverlib\fpu.h" //Supports floating point

#ifndef M_PI
#define M_PI 3.14159265358979323846
#endif

#define SERIES_LENGTH 1000 //depth of buffer

float gSeriesData[SERIES_LENGTH];
int32_t i32DataCount = 0; //Counter for computation loop

int main(void)
{
	float fRadians; //Used to calculate sine
	float fRadians1; //Used to calculate cosine

	ROM_FPULazyStackingEnable();
	ROM_FPUEnable(); //From reset this is off
	ROM_SysCtlClockSet(SYSCTL_SYSDIV_4 | SYSCTL_USE_PLL | SYSCTL_XTAL_16MHZ |
	SYSCTL_OSC_MAIN); //50Mhz
	fRadians = ((2 * M_PI * 50) / SERIES_LENGTH);
	fRadians1 = ((2 * M_PI * 200) / SERIES_LENGTH);

	while (i32DataCount < SERIES_LENGTH)
	{
		gSeriesData[i32DataCount] = 1.5 + (1.0*sinf(fRadians * i32DataCount)) +
		(0.5*(cosf(fRadians1 * i32DataCount)));
		i32DataCount++;
	}
	while (1)
	{
	}
}
```

Grading scheme: 30% Coding, 30% Documentation, 40% Execution/Video.