**Karan Vora (kv2154)**
**Real-Time Embedded Systems Assignment 3**

## Problem 1):

### A)

```
#include <stdint.h>
int32_t DemoArray[20];
for(i = 0; i < 20; i++)
{
    DemoArray[i]=i;
}
```

### B)

```
int even_sum( int *DemoArray, int ArrayLength)
{
    int sum = 0, i;
    for(i = 0; i < ArrayLength; i++)
    {
        if(i%2 == 0)
        {
            sum += DemoArray[i];
        }
    }
    return sum;
}
```

==========================================================================

## Problem 2):

```
int even_sum( int *DemoArray, int ArrayLength)
{
            int sum = 0, i;
            for(i = 0; i < ArrayLength; i++)
            {
                        if(i%2 == 0)
                        {
                                    sum += DemoArray[i];
                        }
                        else
                        {
                                    DemoArray[i]=0;
                        }
            }
            return sum;
}
```

## Problem 3):

```
#include <stdio.h>
#include <stdint.h>

int even_sum(int pointer, int array);

int main()
{
struct DemoStruct
{
unsigned int Integer1, Integer2, Integer3, Integer4;
};

struct DemoStruct DemoStructArray[5];
int i;

for(i = 0; i < 5; i++)
{
DemoStructArray[i].Integer1 = i;
DemoStructArray[i].Integer2 = i+2;
DemoStructArray[i].Integer3 = i+4;
DemoStructArray[i].Integer4 = i+6;
}
//Print The Struct

for(i = 0; i < 5; i++)
{
printf("First element on index %d of DemoStruct Array = %u\n", i, DemoStructArray[i].Integer1);
printf("First element on index %d of DemoStruct Array = %u\n", i, DemoStructArray[i].Integer2);
printf("First element on index %d of DemoStruct Array = %u\n", i, DemoStructArray[i].Integer3);
printf("First element on index %d of DemoStruct Array = %u\n\n", i, DemoStructArray[i].Integer4);
}
return 0;
}
```

=======================================================================

## Problem 4):

```
struct DemoStruct
{
unsigned int Integer1=1;
unsigned int Integer2=2;
unsigned int Integer3=3;
unsigned int Integer4=4;
};
```

```
void zero_structs(DemoStruct *DemoStructArray , int ArrayLength)
{
        int i;
        for(i = 0; i < ArrayLength / 4; i++)
        {
                DemoStructArray->Integer1=0;
                DemoStructArray->Integer2=0;
                DemoStructArray->Integer3=0;
                DemoStructArray→Integer4=0;
                DemoStructArray++
        }
}
```

============================================================================

**Problem 5):**

```
struct DemoStruct{
unsigned int Integer1=1;
unsigned int Integer2=2;
unsigned int Integer3=3;
unsigned int Integer4=4;
};
void fill_structs( DemoStruct *DemoStructArray , int *PointerToArray, int StructArrayLength)
{
   int i = 0;
   for(i = 0; i < StructArrayLength; i++)
   {
      DemoStructArray->Integer1 = *PointerToArray++;
      DemoStructArray->Integer2 = *PointerToArray++;
      DemoStructArray->Integer3 = *PointerToArray++;
      DemoStructArray->Integer4 = *PointerToArray++;
      DemoStructArray++
   }
}
```