Functions – Practice Problems

What are 4 ways to declare user defined function?

Function with no arguments and no return value

```
void displayFixedText()
{

printf("============\n");
}
```

```
int main(void)
{
   displayFixedText();


   displayFixedText();


   return 0;
}
```

Function with no arguments and a return value

```
int a[5]={1, 2, 3, 4, 5};
int n=3; // what is the scope of n?

int find()
{       int i;   // scope of i?
        for (i=0; i<5; i++)
                if (n == a[i]) return 1;
        return 0;
}
```

```
int main(void)
{
   if (find())
      printf("value found");
   else
      printf("value not found");
   return 0;
}
```

Function with arguments and no return value

```
int a[5]={1, 2, 3, 4, 5};
void find(int n) // what is the scope of n?
{       int i;
        for (i=0; i<5; i++)
          if (n == a[i]) {
                  printf("Value found");
                  return;
          }
          printf("value not found");
}
```

```
int main(void)
{
   find(3);
   return 0;
}
```

Function with arguments and a return value

```
int a[5]={1, 2, 3, 4, 5};
int find(int n)
{       int i;
        for (i=0; i<5; i++)
                if (n == a[i]) return 1;
        return 0;
}
```

```
int main(void)
{
    if (find(3))
        printf("value found");
    else
        printf("value not found");
    return 0;
}
```

Problem 1

- Write a function to print_even_numbers to print even numbers between the given two numbers. Answer these questions:
  - Does your function have any arguments passed?
  - Does your function have any return value?

Problem 1 Solution

```
void print_even_numbers(int n1, int n2)
{       int i;
        for(i=n1%2 ? n1+1 : n1; i<=n2; i=i+2)
                printf("%d", i);
}
```

Problem 2

- Write a function power(m, n) which will provide m power n (m^n)
- Now write a function that will calculate following series using power function $1 + x + x^2 + x^3 + \ldots + x^n$

Problem 2 Solution

```
int power(int m, int n)
{
    int i,pow=1;
        for(i=n;i>0;i--)
                pow=pow*m;
    return pow;
}
```

```
int poly_series(int x, int n)
{
    int sum=0,i;
    for(i=0;i<=n;i++)
        sum = sum + power(x,i);
    printf("Series sum %d",sum);

}

int main(void) {  poly_series(2, 3); }
```

Problem 3

- Write a function to calculate the factorial of a number
- Using power function that you wrote earlier and factorial function calculate sin(x) using
- $\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \ldots$

Problem 3 solution

```
int fact(n)
{       int i,prod=1;
        for(i=n;i>0;i++)
                prod=prod*i;
        return prod;
}
```

Use of Static Variable – For next excercise

```
void test()
{
    static int i=0;
    i++;
    printf("%d\n", i);
}
```

```
int main(void)
{
    test();
    test();
    test();
    test();

}
```

Problem 4

- Write a recursive function for summing up all digits of a given number. Note: Recursive functions calls themselves until terminating condition is met.

Problem 4 solution

```
int sum_digits(int n)
{
   static int sum=0;
   if (n > 0)
   {
      sum += n%10;
      sum_digits(n/10);
   }
   else
      return sum;
}
```

```
int main(void)
{
   printf("sum is
%d",sum_digits(1234));
}
```