Task-1

```
#include <stdio.h>
int factorial(int n) {
   if (n == 0) {
      return 1;
   } else {
      return n * factorial(n - 1);
   }
}
int main() {
   int n = 5;
   printf("Factorial of %d is %d", n, factorial(n));
   return 0;
}
```

Task-2

```
#include <stdio.h>

int fibonacci(int n) {
   if (n <= 1) {
      return n;
   } else {
      return fibonacci(n - 1) + fibonacci(n - 2);
   }
}
int main() {
   int n = 5;
   printf("Fibonacci number at position %d is %d", n, fibonacci(n));
   return 0;
}
```

Task-3

```
#include <stdio.h>
void printArray(int arr[], int size, int index) {
   if (index == size) {
      return;
   }
   printf("%d ", arr[index]);
   printArray(arr, size, index + 1);
}
int main() {
   int arr[] = {1, 2, 3, 4, 5};
   int size = sizeof(arr) / sizeof(arr[0]);
   printArray(arr, size, 0);
   return 0;
}
```

Task-5

```
#include <stdio.h>
int power(int m, int n) {
   if (n == 0) {
      return 1;
   } else {
      return m * power(m, n - 1);
   }
}
int main() {
   int m = 5;
   int n = 4;
   printf("Power of %d raised to %d is %d\n", m, n, power(m, n));
   return 0;
}
```