```haskell

fib :: Int -> Int
fib n
    | n < 2     = n
    | otherwise = fib (n - 1) + fib (n - 2)

main :: IO ()
main = do

    print [fib n | n <- [0..19]]
```

```
[1 of 1] Compiling Main             ( main.hs, main.o )
Linking a.out ...
[0,1,1,2,3,5,8,13,21,34,55,89,144,233,377,610,987,1597,2584,4181]
```

```haskell
asterisk_pat :: Int -> Int -> IO ()
asterisk_pat depth space
  | depth <= 0 = return ()
  | otherwise = do
      putStr (replicate_spaces space)
      putStr (replicate_stars depth)
      putStrLn ""
      asterisk_pat (depth - 1) (space + 1)

replicate_spaces :: Int -> String
replicate_spaces n = replicate n ' '

replicate_stars :: Int -> String
replicate_stars n = replicate n '*'

main :: IO ()
main = asterisk_pat 5 0
```

```
[1 of 1] Compiling Main             ( main.hs, main.o )
Linking a.out ...
*****
 ****
  ***
   **
    *
```

```
17
18  % Question 5 kth digit
19  kth_digit(K, N) ->
20    if
21      K >= 4 ->
22        0;
23      K == 1 ->
24        N rem 10;
25      true ->
26        kth_digit(K-1, N div 10)
27      end.
28
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
{ok,final}
9> final:kth_digit(2,32).
3
10> final:kth_digit(2,324).
2
11> final:kth_digit(1,324).
4
12> final:kth_digit(4,324).
0
```

```
18
19  % Quesiton 6 display e
20  find_e(N, X) ->
21      case N of
22          1 -> 1;
23          _ -> pow(X, N-1) / fact(N-1) + find_e(N-1, X)
24      end.
25
26  fact(N) ->
27    case N of
28          0 -> 1;
29          1 -> 1;
30          _ -> N * fact(N - 1)
31    end.
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS F:\3 School\ecs 140> erl
Eshell V13.2  (abort with ^G)
1> c(final).
{ok,final}
2> final:find_e(10, 1).
2.7182815255731922
3> final:find_e(10, 3).
20.063392857142855
```