Computer Programming Paradigm Lab
Lab Experiment No. 6

Name: Shashwat Tripathi Roll. No.: 60
Batch: C Div: D10A

Problem Statement & Output:

Write a Haskell program to

1.Consider a function safetail that behaves in the same way as tail, except that safetail maps the empty list to the empty list, whereas tail gives an error in this case. Define safetail using:

(a) a conditional expression;

(b) guarded equations;

(c) pattern matching.

1.Conditional Expression:

```haskell
safetail xs = if null xs
then []
else tail xs
```

2.Guarded Equations:

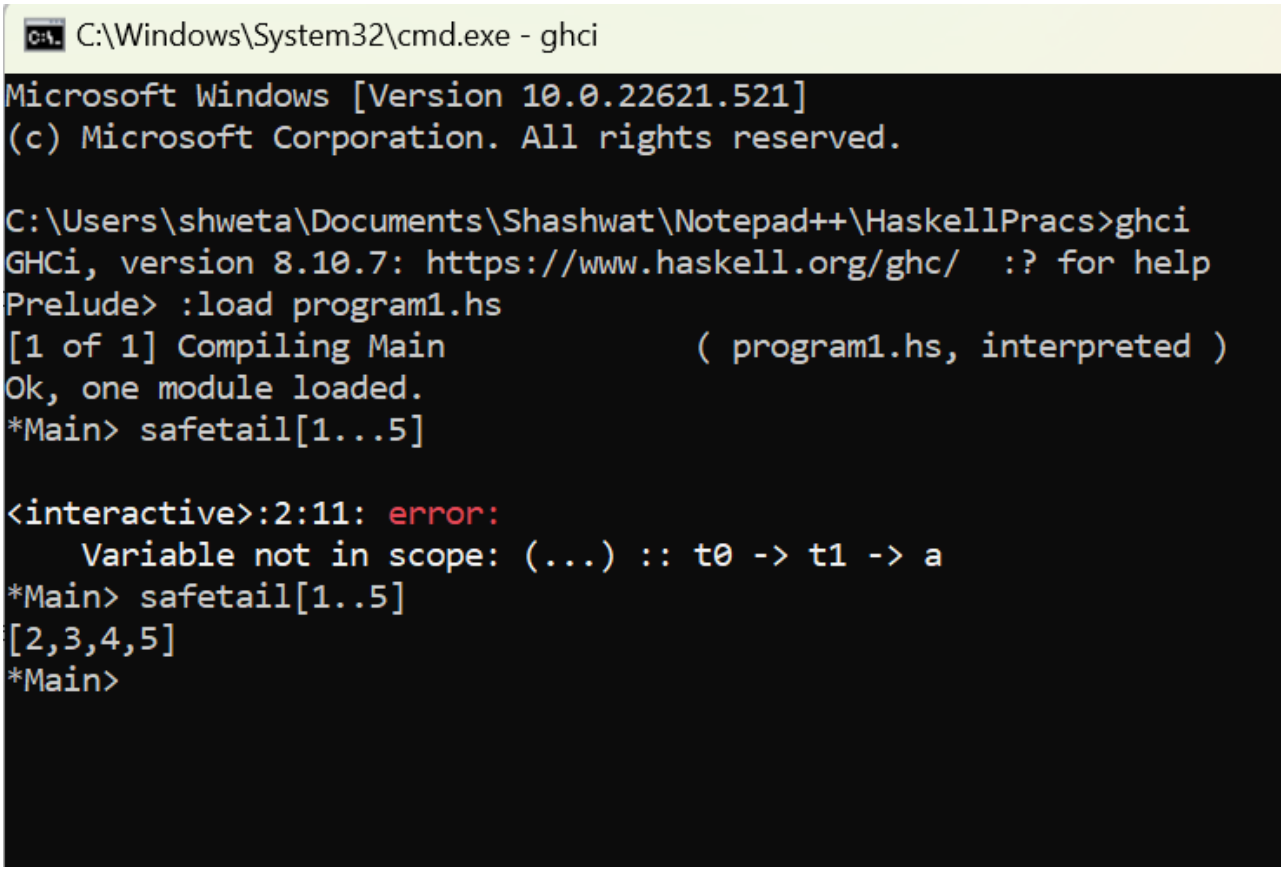
```haskell
safetail xs | null xs = []
| otherwise = tail xs
```

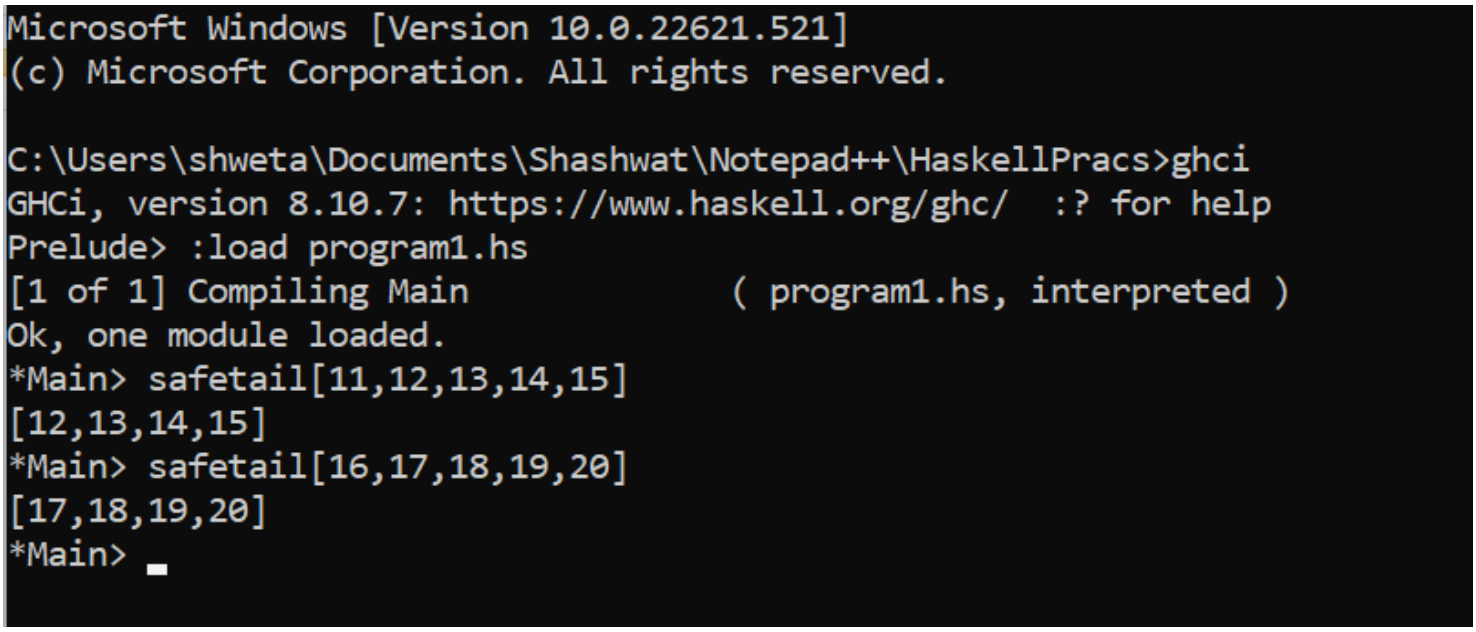
3.Pattern Matching:

```haskell
safetail[] = []
safetail xs = tail xs
```

```
Microsoft Windows [Version 10.0.22621.521]
(c) Microsoft Corporation. All rights reserved.

C:\Users\shweta\Documents\Shashwat\Notepad++\HaskellPracs>ghci
GHCi, version 8.10.7: https://www.haskell.org/ghc/  :? for help
Prelude> :load program1.hs
[1 of 1] Compiling Main             ( program1.hs, interpreted )
Ok, one module loaded.
*Main> [1,2,3,4,5,6,7]
[1,2,3,4,5,6,7]
*Main> safetail[1,2,3,4,5,6,7]
[2,3,4,5,6,7]
*Main> null[]
True
*Main>
```


C:\Windows\System32\cmd.exe - ghci
Microsoft Windows [Version 10.0.22621.521]
(c) Microsoft Corporation. All rights reserved.
C:\Users\shweta\Documents\Shashwat\Notepad++\HaskellPracs>ghci
GHCi, version 8.10.7: https://www.haskell.org/ghc/ :? for help
Prelude> :load program1.hs
[1 of 1] Compiling Main (program1.hs, interpreted)
Ok, one module loaded.
*Main> safetail[1...5]
<interactive>:2:11: error:
Variable not in scope: (...) :: t0 -> t1 -> a
*Main> safetail[1..5]
[2,3,4,5]
*Main>


Microsoft Windows [Version 10.0.22621.521]
(c) Microsoft Corporation. All rights reserved.
C:\Users\shweta\Documents\Shashwat\Notepad++\HaskellPracs>ghci
GHCi, version 8.10.7: https://www.haskell.org/ghc/ :? for help
Prelude> :load program1.hs
[1 of 1] Compiling Main (program1.hs, interpreted)
Ok, one module loaded.
*Main> safetail[11,12,13,14,15]
[12,13,14,15]
*Main> safetail[16,17,18,19,20]
[17,18,19,20]
*Main>

PROGRAM 2:

2. To implement a simple calculator. (use case statement)

```
solveEqn = do
    putStrLn "Enter 1st number"
    input1 <-getLine
    putStrLn "Enter 2nd number"
    input2 <-getLine
    let a =(read input1 :: Int)
    let b =(read input2 :: Int)
    putStrLn "Enter the operator from ('+','-','/','*')"

    oper<-getChar
    return(case oper of
        '+'-> (a+b)
        '-'-> (a-b)
        '/'-> (a`div`b)
        '*'-> (a*b))

main = do
eval <- solveEqn
print("Answer is: " ++show(eval))
```

```
Microsoft Windows [Version 10.0.22621.521]
(c) Microsoft Corporation. All rights reserved.

C:\Users\shweta\Documents\Shashwat\Notepad++\HaskellPracs>ghci
GHCi, version 8.10.7: https://www.haskell.org/ghc/  :? for help
Prelude> :load program2.hs
[1 of 1] Compiling Main             ( program2.hs, interpreted )
Ok, one module loaded.
*Main> main
Enter 1st number
729
Enter 2nd number
9
Enter the operator from ('+','-','/','*')
/
"Answer is: 81"
*Main>
```