METU NCC, CNG 242 Programming Language Concepts
Spring 2010, Haskell Quiz
Open Book and notes, 50 minutes, 50 points, **??** questions, **??** pages

Name, Surname: _____ **Id:** _____

QUESTION 1. (15 points)

What are the values of the following Haskell expressions:

```
let x = 3
    y = 2
    z = 1
in  let x = 1
    in x+y+z          [        ]
```

```
let x = 3
    y = 2
    z = 1
in  let y = 1
    in x+let x=5
           in x+y+z    [        ]
```

```
let f x = x+1
    g x = x*2
in  f (let f x = x*x in g(f 3))+f 1    [        ]
```

```
let tk n (x:xs) | n==0 = []
                | otherwise = x:(tk (n-1) xs)
in tk 3 [1,5,2,3,4,0,1]    [        ]
```

```
let mg a [] = a
    mg [] a = a
    mg (x:xs) (y:ys) = if x<y then  x:(mg xs (y:ys))
                              else y:(mg (x:xs) ys)
in mg [1,5,4] [3,2]    [        ]
```

```
let m n x = case x of
              [] -> n==0
              (_:r) -> m (n-1) r
in m 3 [1,2,3,4,5]    [        ]
```

Assume the following function definitions exist:

```
nmap f [] = []
nmap f (x:xs) = (f x):(nmap f xs)

nfilter f [] = []
nfilter f (x:xs) = if (f x) then (x:(nfilter f xs))
                   else (nfilter f xs)

reduce f s [] = s
reduce f s (x:xs) = f x (reduce f s xs)

for m n f s = if (m>n) then s
                 else for (m+1) n f (f m s)

comp f g x y = f x (g x y)

iter 0 f s = s
iter n f s = iter (n-1) f (f s)

zpt a [] = [a]
zpt a (x:xs) = if a>x then a:(x:xs)
               else x:(zpt a xs)
```

What are the values of the following expressions?:

nmap even [1,2,3,4,5,6]

[False,True,False,True,False,True]

nmap ((+) 2) (nfilter even [1,2,3,4,5,6])

[4,6,8]

iter 10 ((*) 2) 4

4096

reduce zpt [] [1,3,4,2,3,1]

[4,3,3,2,1,1]

for 10 13 zpt []

[13,12,11,10]

reduce (comp (*) (+)) 1 [1,2,3]

29

Note: (*) x y = x*y ;   (+) x y = x+y ;   even x = (x 'mod' 2 == 0)

QUESTION 3. (15 points)

a) Complete the following function to return number of occurences of the first parameter in the second parameter list. occurs 1 [1,2,3,1,2] will return 2 since 1 has 2 occurrences in the list. occurs 3 [1,2,3,1,2] will return 1 and occurs 4 [1,2,3,1,2] will return 0.

```
occurs _ [] = 0
occurs a (x:xs) = let rest=occurs a xs
                  in if a==x then [        ]

                       else [        ]
```

b) Complete the following function to return number of occurences of the first parameter in the nodes of the second parameter tree. The definition of tree is given below. In the sample tree t the calls return the corressponding values:

occurst 1 t → 6
occurst 2 t → 2
occurst 3 t → 1
occurst 4 t → 0

```
data Tree a = Node (a, Tree a, Tree a) | Empty deriving Show

occurst _ Empty = 0
```

```
occurst a (Node (x, left, right)) =  [                    ]
```

```
t = Node (3,Node (2,Node (1,Empty,Empty),Node (1,Empty,Empty)),
          Node (1,Node (2,Node (1,Empty,Empty),Node (1,Empty,Empty)),
                Node (1,Empty,Empty)))
```