Practise quiz (MATH2504, 2025)

Question 1:

What is the output of the following code?

In [ ]:
```
i=94
while i>1
    i=i ÷ 3
    println(i)
    if i<20
        break
    end
end
```

Question 2:

What is the output of the following code?

In [ ]:
```
booleanoperator(cond1::Bool, cond2::Bool)=(cond1 || cond2) && (cond1 != cond2)
count=0
for i=1:100
    if booleanoperator(i<50, i<80)
        count += 1  # this is the same as count=count+1
    end
end
@show count
```

Question 3:

Fix the bug in the following code

In [19]:
```
function normalisevec!(vec::Vector{Float16})
    maxnorm=maximum(abs, vec)
    vec=vec/maxnorm



end
v=randn(Float16, 6)  #vector of normally distributed random numbers
println("Vector v before normalisation:\t\t\t", v)
normalisevec!(v)
println("After normalisation (max norm equal to 1):\t", v)
```

```
Vector v before normalisation:                  Float16[-0.2676, 0.5986, -0.06537,
-1.937, -0.295, -0.2942]
After normalisation (max norm equal to 1):      Float16[-0.2676, 0.5986, -0.06537,
-1.937, -0.295, -0.2942]
```

Question 4:

Formulate a function which finds the position of the first and the last occurance of a given letter in a given string. Distinguish between upper and lower case letters. Add the missing two lines in the code.

In [ ]:
```
function findletters(str::String, c::Char)
    posfirst=0
    poslast=0
    for i in 1:length(str)
        if str[i]==c

            ..................................
                posfirst = i

            ..................................
            poslast = i
        end
    end
    return (posfirst, poslast)
end
findletters("Partial Differential Equations", 'a')
```

Question 5:

Now, add the missing lines in a function which computes the number of occurances of a given letter in a string.

In [ ]:
```
function countletters(str::String, c::Char)

    ......................
    for i in 1:length(str)
        if str[i]==c

            ..................................

        end
    end

    ..........................
end
countletters("Partial Differential Equations", 'a')
```

Question 6:

The following function is meant to find all the prime numbes between 1 and n. However, the function has 3 bugs. Find and fix them all.

In [ ]:
```
function findprimes(n::Int)
    isprime=fill(false, n)
    for testnum=2 : n
        if isprime[testnum]
            for j = 2 : (n ÷ testnum)
                isprime[testnum]=false
            end
        end
    end
    primes=Int[]
    for i=2:n
        if isprime(i)
            push!(primes, i)
        end
    end
    return primes
end
findprimes(44)
```