

Name(n):

eigentlich kontrollierende(r) Tutor(in) (Postfach):

Aufgabe 1:

Gegeben ist folgender Code:

```
function ausgabe = fkt(n)
ausgabe = '';
if n == 0
    ausgabe='0';
end
while n>0
    st=mod(n,2);
    ausgabe=[num2str(st), ausgabe];
    n=floor(n/2);
end
```

1. Beschreiben Sie in einem Satz, was das Programm macht!
2. Welche Ausgabe erhalten Sie bei dem Aufruf *fkt(27)*? (Tipp: Geben Sie ggf. **nur zur Kontrolle** das Programm in Matlab ein!)

Aufgabe 2:

Gegeben ist folgender Code:

```
1
2 function uebung(a,b)
3     x = [ ]
4     n = 0;
5     ende = a*b;
6
7     if a<b
8         c = a;
9         a = b;
10        b = c;
11    end
12
13    for k=a:ende
14        if (mod(k,a)==0 && mod(k,b)==0)
15            n = n + 1;
16            x(n) = k;
17        end
18    end
19
20    disp(x)
21 (Ende des Programms)
```

1. Wozu dient die if-Abfrage? Also was passiert dort und warum ist sie notwendig?
2. Welche Ausgabe liefert *uebung(10,15)*?

Aufgabe 3:

Gegeben ist ein fehlerhafter Code! Ohne Fehler rechnet das Programm die einzugebene Zahl n in eine beliebige Basis q um. Als Output erhält man einen Vektor z mit den Ziffern $(1, \dots, q-1)$ als Einträge.

```
1 function z = dec2basis(n)
2
3 if (n=0)
4     erg = 0;
5 else
6     erg = [ ];
7 end;
8
9 while (n=0)
10     erg = [mod(n,q), erg];
11     n = floor(n/q);
12 end
13
14 z = erg
```

Finden Sie alle **fünf** Syntax-Fehler und geben Sie an, wie diese ausgebessert werden müssen!
Hinweis: Zeile 2, 8 und 13 sind wegen der Übersichtlichkeit leer!

Aufgabe 4:

Gegeben sei folgender Code:

```
p = input('Bitte geben Sie eine natürliche Zahl größer als 2 ein: ');
v = [];
for i = 2:p-1
    if mod(p,i)==0
        v = [v,i];
    else
        v = v;
    end
end
if length(v) == 0
    disp('p ist ...')
else
    v
end
end
```

1. Was ermittelt der Code?
2. Ergänzen Sie die Textausgabe 'p ist ...' sinnvoll!
3. Es sei das Aufwandsmaß definiert durch die Anzahl der Schleifendurchläufe. Optimieren Sie den Aufwand des Programms hinsichtlich der Laufzeit!

Aufgabe 5:

Als Grundlage für die in der Vorlesung behandelte LR-Zerlegung soll in dem folgenden Code eine Gauß-Umformung vorgenommen werden. Dabei wird eine $n \times n$ -Matrix A in eine obere Dreiecksmatrix R umgeformt. Allerdings ist dem Autor des Codes ein logischer Fehler unterlaufen.

1. Beheben Sie diesen Fehler!
2. Bauen Sie eine Sicherheitsschranke ein, die das Programm *inkl. Fehlermeldung abbricht*, wenn A keine quadratische Matrix ist!
3. Betrachten Sie die Eingabe `carlf([3,4,2;5,8,5;1,2,3])`. Welchen Wert haben die Matrizen L und R zum Schluss?

```
function [L,R] = carlf(A)

[m,n] = size(A);
L = eye(m);
R = A;
for i = 1:m-1
    for k = i+1:m
        L(k,i) = R(k,i)/R(i,i);
        for j = i+1:n
            R(k,j) = R(k,j) - L(k,i)* R(i,j);
        end
        for j = 1:i
            R(k,j) = 1;
        end
    end
end
```