



HAUSAUFGABEN

30.09.2024

Inhaltsverzeichnis

1. Skript 1: Grösser als 10	2
1.1. Erklärung	2
1.2. Test	2
1.3. Diagramme	3
1.3.1. Programmablaufplan (PAP)	3
1.3.2. Nassi-Shneidermann Diagramm (NSD)	3
2. Skript 2: Gerade oder ungerade Zahl	4
2.1. Erklärung	4
2.2. Test	4
2.3. Diagramme	5
2.3.1. Programmablaufplan (PAP)	5
2.3.2. Nassi-Shneidermann Diagramm (NSD)	5
3. Skript 3: Passwort Checker	6
3.1. Erklärung	6
3.2. Test	7
3.3. Diagramme	7
3.3.1. Programmablaufplan (PAP)	7
3.3.2. Nassi-Shneidermann Diagramm (NSD)	8

1. Skript 1: Grösser als 10

1.1. Erklärung

Im Skript «greater_than_10.sh» wird überprüft, ob eine Zahl grösser, kleiner oder genau 10 ist.

```
#!/bin/bash

# Hier fragt der Skript nach eine Zahl welche du prüfen möchtest.
read -p "Bitte gebe eine Zahl ein: " number

# Hier wird überprüft ob die Zahl grösser, kleiner oder genau 10 ist.
if [ $number -gt 10 ]
then
    echo "Die Zahl ist grösser als 10."
elif [ $number -lt 10 ]
then
    echo "Die Zahl ist kleiner als 10."
else
    echo "Die Zahl ist genau 10."
fi
```

Als erstes fragt das Programm nach einer Zahl, diese wird dann mit der Variable «number» gespeichert.

Danach ruft es die «number» variable auf und überprüft, ob es grosser als 10 ist mit «-gt» welches «greater than» bedeutet.

Ist dies nicht der Fall geht es weiter und überprüft mit «-lt», welches «less than» bedeutet, ob die Zahl kleiner ist als 10.

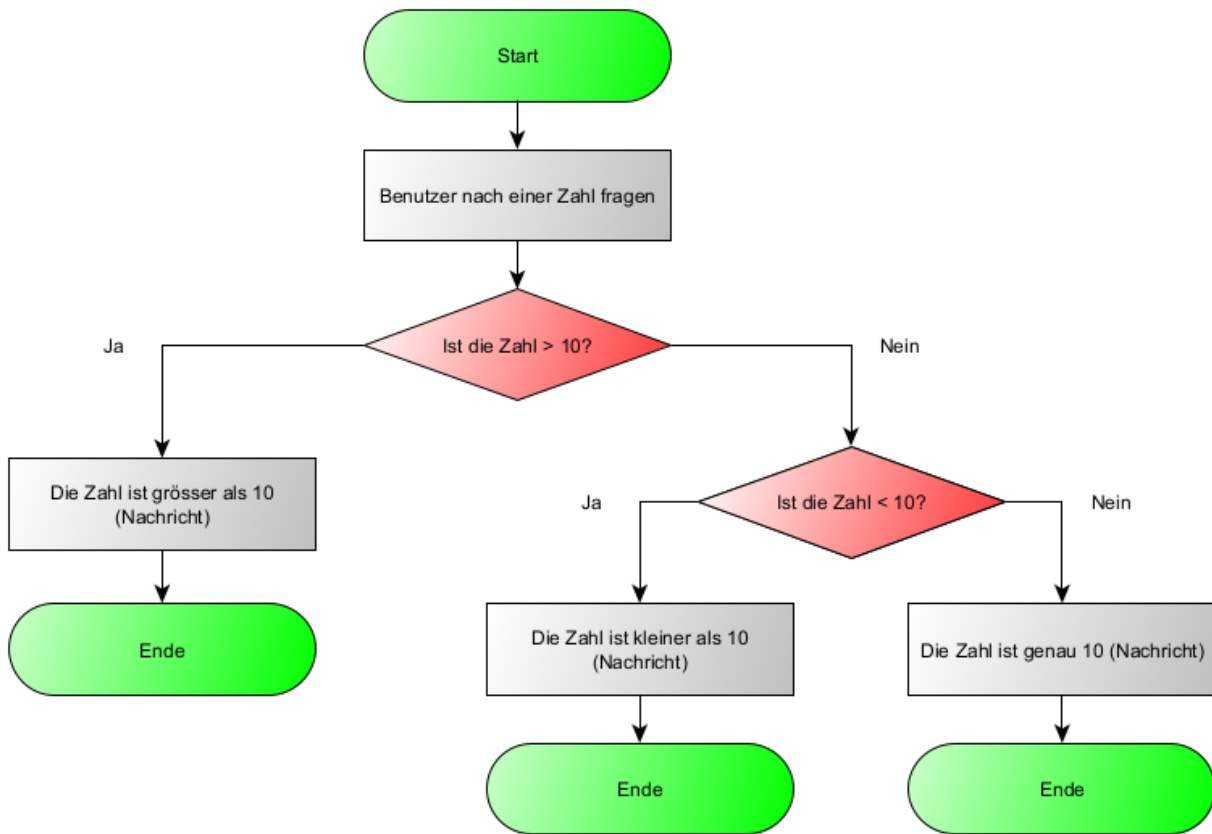
Wenn die Zahl nicht grösser oder kleiner ist als 10 geht es weiter und teilt uns mit das die Zahl genau 10 ist.

1.2. Test

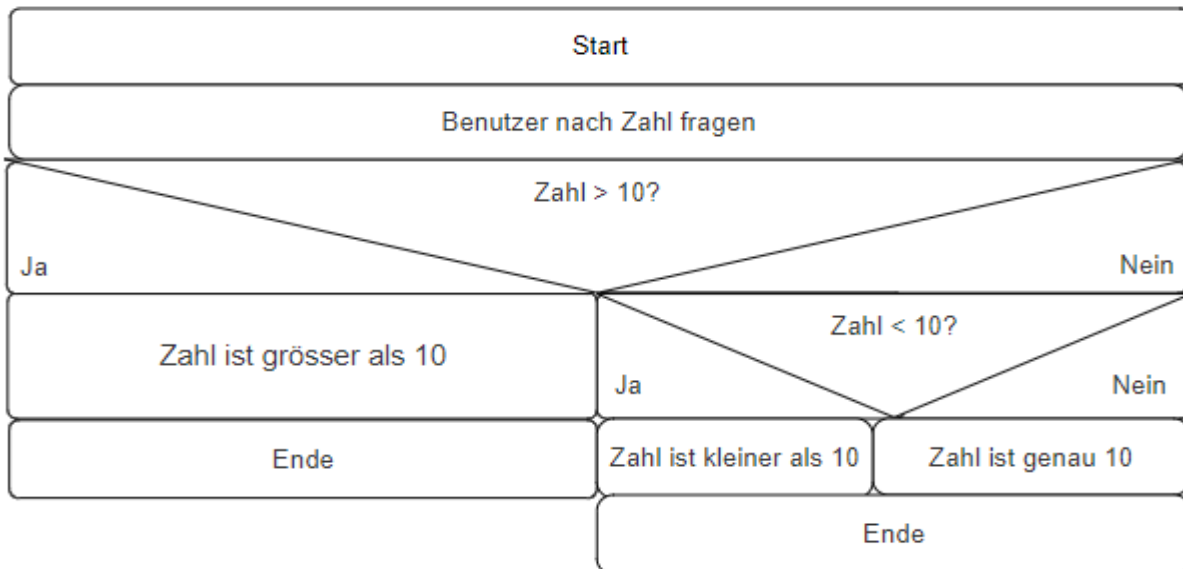
```
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./greater_than_10.sh
Bitte gebe eine Zahl ein: 5
Die Zahl ist kleiner als 10.
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./greater_than_10.sh
Bitte gebe eine Zahl ein: 15
Die Zahl ist grösser als 10.
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./greater_than_10.sh
Bitte gebe eine Zahl ein: 10
Die Zahl ist genau 10.
```

1.3. Diagramme

1.3.1. Programmablaufplan (PAP)



1.3.2. Nassi-Shneidermann Diagramm (NSD)



2. Skript 2: Gerade oder ungerade Zahl

2.1. Erklärung

Im Skript «even_or_odd.sh» wird überprüft, ob eine Zahl gerade oder ungerade ist in dem das Programm überprüft, ob die Zahl durch 2 teilbar ist

```
#!/bin/bash

# Hier fragt es nach deine Zahl
read -p "Bitte gebe eine Zahl ein: " number

# Es wird überprüft ob die Zahl durch 2 teilbar ist,
# ist dies der Fall ist die Zahl gerade.
if [ $(( $number % 2 )) -eq 0 ]
then
| echo "Die Zahl ist gerade."
else
| echo "Die Zahl ist ungerade."
fi
```

Als erstes fragt das Programm nach einer Zahl, diese wird dann mit der Variable «number» gespeichert.

Der «if» Statement teilt die Zahl durch 2 und testet mit «-eq», welches «equal» bedeutet, ob es gleichwertig ist mit 0.

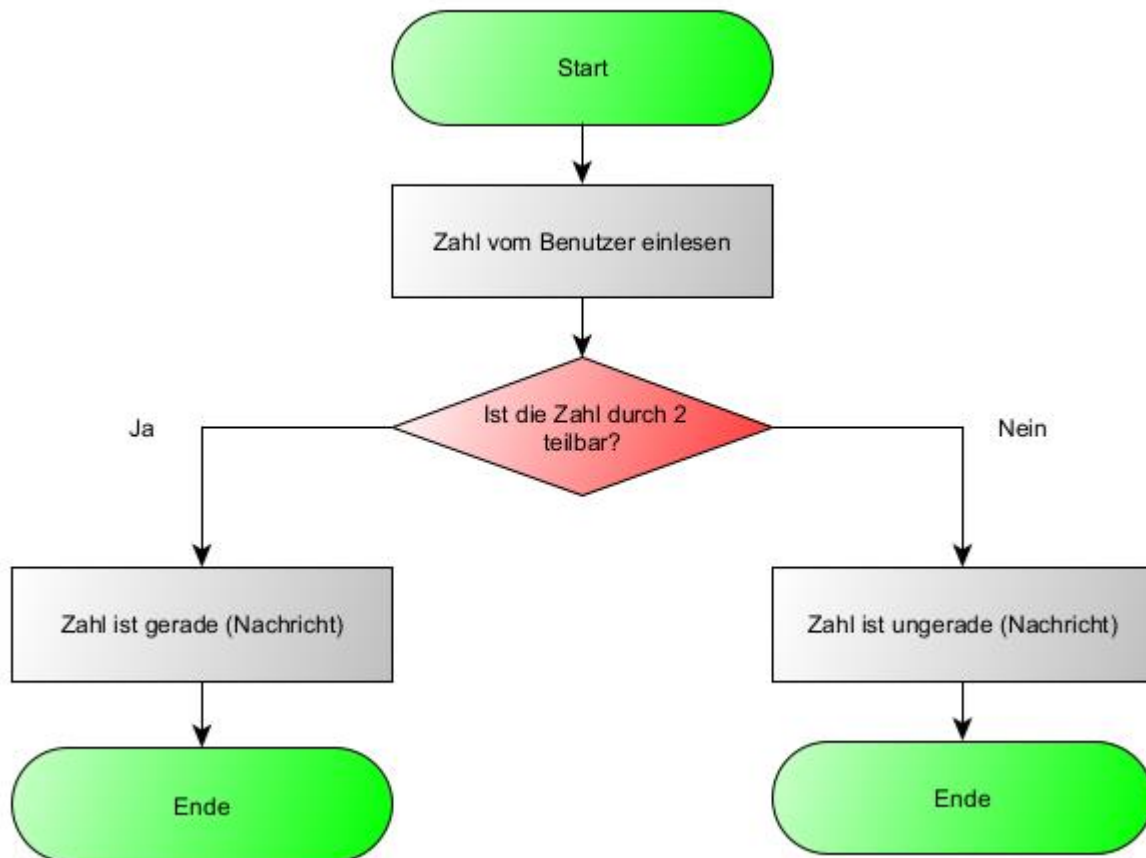
Wenn dies der Fall ist, dann ist die Zahl gerade, wenn nicht ist die Zahl ungerade.

2.2. Test

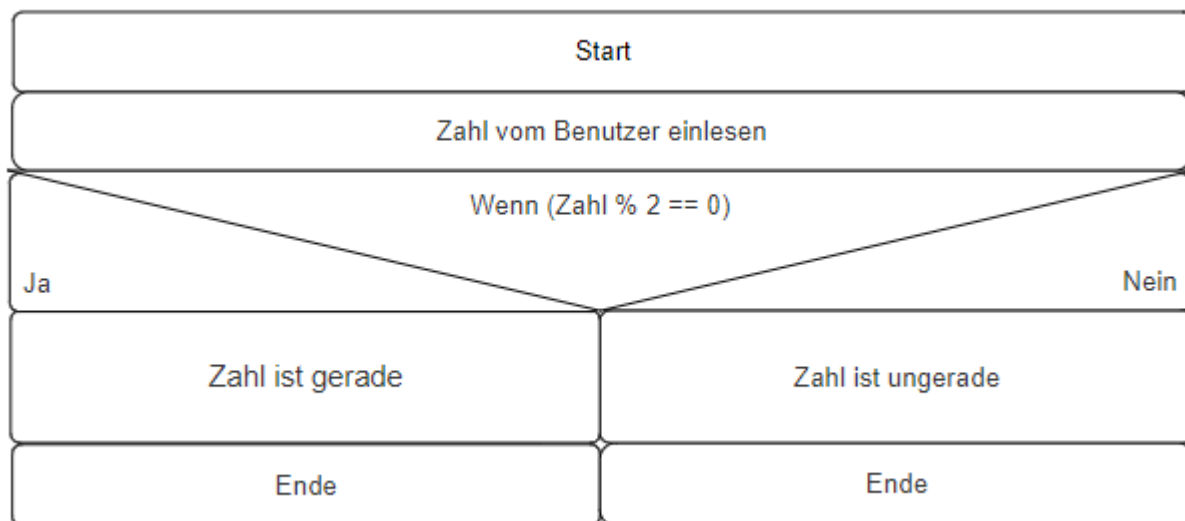
```
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./even_or_odd.sh
Bitte gebe eine Zahl ein: 5
Die Zahl ist ungerade.
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./even_or_odd.sh
Bitte gebe eine Zahl ein: 8
Die Zahl ist gerade.
```

2.3. Diagramme

2.3.1. Programmablaufplan (PAP)



2.3.2. Nassi-Shneidermann Diagramm (NSD)



3. Skript 3: Passwort Checker

3.1. Erklärung

Das Skript «password_checker.sh» überprüft, ob das eingegebene Passwort 8 Zeichen und eine Zahl hat.

```
#!/bin/bash

# Hier gibt man sein Passwort ein.
read -p "Bitte gebe deinen Passwort ein: " password
echo

# Hier prüft es ob das Passwort mindestens 8 Zeichen lang ist.
if [ ${#password} -lt 8 ]
then
    echo "Passwort ist zu kurz, es muss mindestens 8 Zeichen lang sein."
else
    # Prüft ob das Passwort mindestens eine Zahl enthält.
    if [[ "$password" =~ [0-9] ]]
    then
        echo "Passwort ist gültig."
    else
        echo "Passwort muss mindestens eine Zahl enthalten."
    fi
fi
```

Zuerst fragt das Programm nach einem Passwort, diese wird dann mit der Variable «password» gespeichert.

Dann überprüft das Programm, ob das Passwort länger als 8 Zeichen ist, wenn es unter 8 Zeichen ist, sagt das Programm, dass es mindestens 8 Zeichen lang sein muss.

Dann wird überprüft, ob das Passwort eine Zahl von 0 bis 9 drin hat, wenn nicht wird dies mitgeteilt.

Falls alles erfüllt, ist dann teilt das Programm mit, dass das Passwort gültig ist.

3.2. Test

```
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./password_check.sh
Bitte gebe deinen Passwort ein: abc

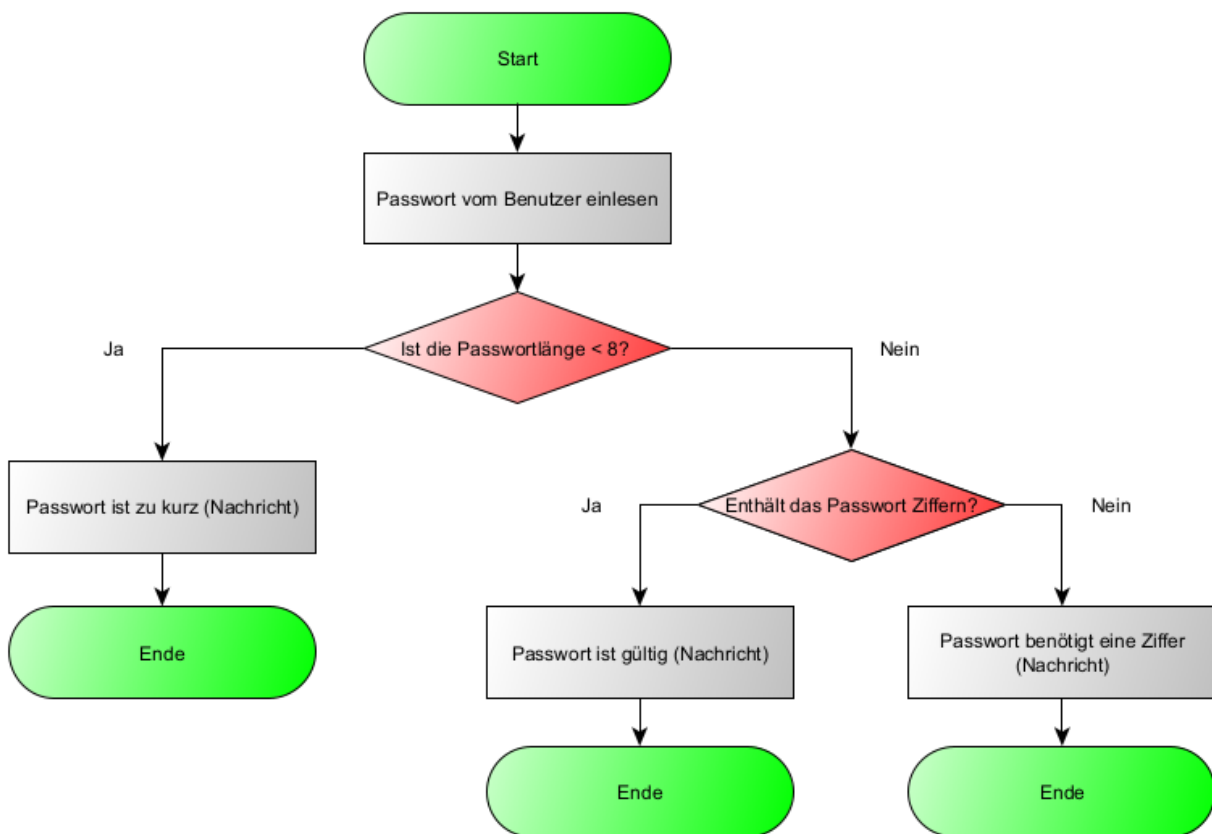
Passwort ist zu kurz, es muss mindestens 8 Zeichen lang sein.
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./password_check.sh
Bitte gebe deinen Passwort ein: abcdefgh

Passwort muss mindestens eine Zahl enthalten.
nikola@nikola-OdinVMware:~/Documents/repos/BashProjekte/HA_30-9$ ./password_check.sh
Bitte gebe deinen Passwort ein: abcdefgh1

Passwort ist gültig.
```

3.3. Diagramme

3.3.1. Programmablaufplan (PAP)



3.3.2. Nassi-Shneidermann Diagramm (NSD)

