

**<20Gewinnt Bash>**

|  |  |
| --- | --- |
| NAME, VORNAME | Bischof, David |
| KLASSE | **5IM21a** |

|  |  |
| --- | --- |
| LEHRPERSON | **J. Vogel** |
| ABGABETERMINE | **04.11.2022** |

Inhalt

[0 Einleitung 2](#_Toc118447642)

[1 Strucktogramm zum 1 gegen 1 2](#_Toc118447643)

[2 Reflektion 3](#_Toc118447644)

[3 Scripts 4](#_Toc118447645)

[3.1 Start.sh 4](#_Toc118447646)

[3.2 20gewinnt\_2player.sh 5](#_Toc118447647)

[3.3 20gewinnt\_random.sh 7](#_Toc118447648)

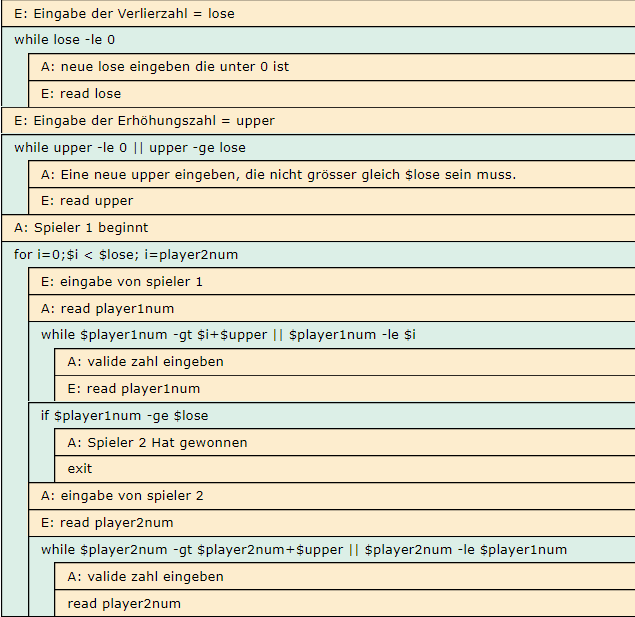
[3.4 20gewinnt\_winBot.sh 9](#_Toc118447649)

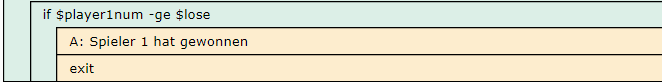
# Einleitung

*Im Fach M122 wurde uns der Auftrag gegeben, ein Programm mit Bash zu erstellen, welches das Spiel 20Gewinnt abspielen kann.*

*Ich habe die Aufgabe mit Variablen gelösst, sodass man selbst bestimmen kann um wie viel man erhöhen will und auf welcher Zahl man verliert/gewinnt.*

# Strucktogramm zum 1 gegen 1

**

**

*Das ist das Stucktogramm für mein Bash Script. Ich habe mein Struktogramm so dargestellt, da Herr Vogel mir gesagt hat, mein vorheriges Struktogramm sei schlecht gewesen.*

# Reflektion

*Ich habe mir in der estern Lektion gedanken gemacht, wie ich das Spiel verändern kann, dass es einen «Wiederspiel-Wert» hat. Meine Finale Idee war es, dass man verschiedene Zahlen eingeben kann und nicht nur 20 und 2.*

*Danach habe ich mir gedanken gemacht, wie ich einen Bot erstellen kann, welcher immer gewinnt. Ich habe eine simple Formel gefunden, bei welcher der Bot immer gewinnen wird.*

*Beim Scripten hatte ich zwar noch mühe mit den Abständen und einige male habe ich falsche Variabeln genutzt aber es war immer noch sehr spassig dieses erste Script zu erstellen, auch wenn es sehr primitiv ist (noch einfacher als die anderen zwei).*

*Dieses erste Script habe ich komplet alles auf einer Mulitpass VM in einem Terminal gecodet. Nachdem ich fertig war mit diesem Script wollte ich mich eigentlich den Zwei andern Modis widmen, jedoch ist plötzlich meine VM kaputt gegangen und jeglicher Fortschritt schien verloren. In den Ferien habe ich das Problem jeodch wieder beheben können, ohne dass ich weiss wie.  
  
Nach diesem Rückfall habe ich mein Scrip von der VM hinüber in VisualStudio kopiert, dass solche Probleme nicht wieder Vorfallen. In VS hat alles sehr gut funktioniert und ich habe meine beiden Scripts ohne grosse Probleme vervollständigen können.*

*Als ich fertig war habe ich Leonard meine Scripts zum testen gegeben und er hat sie einwenig zerstören können. Diese Bugs habe ich danach gefixt und er konne mein Script nicht mehr kaputt machen.*

*Obwohl Bash eher eine einfach sprache ist, macht es sehr viel Spass ein solches Script zu Coden, da man immer neue Sachen herausfindet, wie zum Beispiel, wie man eine Random Zahl erstellt.*

# Scripts

## Start.sh

#!/bin/bash

echo  "Regeln:"

echo  "Dies ist ein Zählspiel, wobei derjenige verliert, der die Verlierzahl sagen muss "

echo  "Mann kann die Zahl um die Vorherige bestimmte Anzahl erhöhen."

echo  "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"

echo  "Es gibt insgesamnt 3 modis, welche man spielen kann"

echo "Beim ersten Modus spielt man gegen einen Bot, welcher einfach random ausgaben macht."

echo "Diesen Bot kann man starten indem man unten: 'Random' eingibt."

echo ""

echo "Beim zweiten Modus spielt man gegen einen Bot der immer perfekt spielt und man nicht besiegen kann."

echo "Diesen Modus kann man spielen, wenn man unten :'Verlieren' eingibt."

echo ""

echo "Beim letzten Modus spieltst du gegen einen freund und machst ein eins gegen eins."

echo "Um diesen Modus zu spielen mussman unten: '1gegen1' eingeben."

echo  "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"

echo -n "Was wilst du Spielen? (Random, Verlieren oder 1gegen1)Eingabe: "

read Modus

#Falls Der Modus Random gewählt wird

if [[ "$Modus" = "Random" ]]

then

    clear

    ./20gewinnt\_random.sh

#Falls Der Modus Verliern gewählt wird

elif [[ "$Modus" = "Verlieren" ]];

then

    clear

    ./20gewinnt\_winBot.sh

#Falls Der Modus 1gegen1 gewählt wird

elif [[ "$Modus" = "1gegen1" ]]

then

    clear

    ./20gewinnt\_2player.sh

else

#falls keine gültige Eingabe getätigt wurde

echo "Das ist keine Richtige eingabe. Das Script ist nun beendet."

fi

## 20gewinnt\_2player.sh

#!/bin/bash

#Bestimmt die Spielregeln

echo  "Bestimmt bitte zuerst untereinander wer Spieler 1 und wer Spieler 2 ist."

#Bestimmt die beiden Zahlen

echo -n "Bitte Verlierzahl eingeben: "

read losenumber

#Checkt, dass die Verlierezahl nicht unter Null ist

while [[ $losenumber -le 0 ]]

 do

    echo  "Die Verliererzahl darf nicht unter Null liegen."

    echo -n "Bitte eine neue Verlierezahl eingeben: "

    read losenumber

 done

echo -n "Bitte erhöhende Zahl eingeben: "

read uppernumber

#Checkt, dass die Erhöhungszahl nicht unter null ist und  nicht über gleich der Verlierzahö

while [[ $uppernumber -lt 1 || $uppernumber -gt $losenumber ]]

 do

    echo "Die erhöhende Zahl muss mindestens 1 und darf nicht über/gleich $losenumber sein."

    echo -n "Bitte eine neue erhöhungs Zahl eingeben: "

    read uppernumber

 done

 echo "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"

#Beginn des Spieles

echo "Spieler 1 beginnt"

for((i=0;$i < $losenumber; i=$player2num)); do

    echo $i

    player1sum=$((i+1))

    player1max=$((i+uppernumber))

    echo -n "Spieler 1 gib eine Zahl von $player1sum bis $player1max: "

    read player1num

    #Wir ausgeführt, falls die eingegebene Zahl nicht erlaubt ist

    while [[ $player1num -lt $player1sum || $player1num -gt $player1max ]]

    do

        echo "Spieler 1 deine Zahl darf nicht unter $player1sum und nicht über $player1max sein!"

        echo -n "Bitte eine Neue Zahl eingeben: "

        read player1num

    done

    #wir ausgeführ falls Spieler 1 eine Zahl eingibt, welche grösser gleich die Verliezahl ist

    if [[ $player1num -ge $losenumber ]]

    then

        echo "Spieler 2 hat gewonnen."

        exit

    fi

    player2sum=$((player1num+1))

    player2max=$((player1num+uppernumber))

    echo "p1 $playe1max"

    echo "p2 max $player2max"

    echo -n "Spieler 2 gib eine Zahl von $player2sum bis $player2max ein:"

    read player2num

        #Wir ausgeführt, falls die eingegebene Zahl nicht erlaubt ist

    while [[ $player2num -lt $player2sum || $player2num -gt $player2max ]]

    do

        echo "Spieler 2 deine Zahl darf nicht unter $player2sum und nicht über $player2max sein!"

        echo -n "Bitte eine neue Zahl eingeben: "

        read player2num

    done

    #wir ausgeführ falls Spieler 2 eine Zahl eingibt, welche grösser gleich die Verliezahl ist

    if [[ $player2num -ge $losenumber ]]

    then

        echo "Spieler 1 hat gewonnen."

        exit

    fi

 done

## 20gewinnt\_random.sh

#!/bin/bash

#Bestimmt die Spielregeln

echo  "Du wirst gegen einen Bot spielen, welcher ohne Strategie und nur random Zahlen ausgibt."

#Bestimmt die beiden Zahlen

echo -n "Bitte Verlierzahl eingeben: "

read losenumber

#Checkt, dass die Verlierezahl nicht unter Null ist

while [[ $losenumber -le 0 ]]

 do

    echo  "Die Verliererzahl darf nicht unter Null liegen."

    echo -n "Bitte eine neue Verlierezahl eingeben: "

    read losenumber

 done

echo -n "Bitte erhöhende Zahl eingeben: "

read uppernumber

#Checkt, dass die Erhöhungszahl nicht unter null ist und  nicht über gleich der Verlierzahl ist

while [[ $uppernumber -lt 1 || $uppernumber -gt $losenumber ]]

    do

    echo "Die erhöhende Zahl muss mindestens 1 und darf nicht über/gleich $losenumber sein."

    echo -n "Bitte eine neue erhöhungs Zahl eingeben: "

    read uppernumber

 done

 echo "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"

#Beginn des Spieles

echo "Spieler 1 beginnt"

for((i=0;$i < $losenumber; i=$bot2insert)); do

    player1min=$((i+1))

    player1max=$((i+uppernumber))

    echo -n "Spieler 1 gib eine Zahl von $player1min bis $player1max: "

    read player1num

        #Wir ausgeführt, falls die eingegebene Zahl nicht erlaubt ist

    while [[ $player1num -lt $player1min || $player1num -gt $player1max ]]

    do

        echo "Spieler 1 deine Zahl darf nicht unter $player1min und nicht über $player1max sein!"

        echo -n "Bitte eine Neue Zahl eingeben: "

        read player1num

    done

    #Wird ausgefürh falls der Bot gewonnen hat

    if [[ $player1num -ge $losenumber ]]

        then

        echo "Der Bot hat gewonnen."

        exit

    fi

    #Der Code vom Bot

    bot2insert=$((1 + $RANDOM % uppernumber + player1num ))

    echo  "Der Computer sagt die Nummer " $bot2insert

    #Wird ausgefürh falls der Spieler gewonnen hat

    if [[ $bot2insert -ge $losenumber ]]

        then

        echo "Du hast gewonnen."

        fi

 done

## 20gewinnt\_winBot.sh

#!/bin/bash

# 20 gewinnt spiel für die 5im21a

# 2022-10-28

#Bestimmt die Spielregeln

echo  "Dieser Bot spielt mit der perfekten Strategie."

#Bestimmt die beiden Zahlen

echo -n "Bitte Verlierzahl eingeben: "

read losenumber

#Checkt, dass die Verlierezahl nicht unter Null ist.

while [[ $losenumber -le 0 ]]

do

    echo  "Die Verliererzahl darf nicht unter Null liegen."

    echo -n "Bitte eine neue Verlierezahl eingeben: "

    read losenumber

done

#nimmt die Erhöhungszahl

echo -n "Bitte erhöhende Zahl eingeben: "

read uppernumber

#Checkt ob die Erhöhungszahl unter der Verlierzahl und über Null ist.

while [[ $uppernumber -le 0 || $losenumber -le $uppernumber ]]

do

    echo "Die erhöhende Zahl muss mindestens 1 sein und weniger als $losenumber."

    echo -n "Bitte eine neue erhöhungs Zahl eingeben: "

    read uppernumber

done

#Erstellt die Variabeln, sodass der Computer nicht verliert

truewin=$((losenumber-1))

trueupper=$((uppernumber+1))

echo "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"

#Das Spiel

start=$((truewin%trueupper))

if [[ $start -ne 0 ]]

then

    echo "Der Computer beginnt"

fi

for((i=$start;$i<=$truewin;i+=$trueupper)); do

    #Ist gezwungen, sodass der BBot immer gewinnen kann

    if [[ $i -eq 0  ]]

    then

        echo -n "Du beginnst. Wähle eine Zahl von 1 bis $uppernumber: "

        read playernumber

    else

        echo "Der Computer sagt die Nummer $i"

        echo -n "Was ist deine Zahl? Maximal plus $uppernumber:"

        read playernumber

    #Wir ausgeführt, falls die eingegebene Zahl nicht erlaubt ist

    while [[ $playernumber -gt $uppernumber+$i || $playernumber -le $i ]]

    do

        maxupper=$((uppernumber+i))

        echo "Hey die Zahl darf nicht kleiner/gleich $i sein oder grösser als $maxupper sein! (Maximal + $uppernumber)"

        echo -n "Bitte eine richtige Zahl eingeben: "

        read playernumber

    done

    #wird ausgefürht sobald der bot gewinnt, was er immer tut.

    if [[ $i+1 -eq $losenumber  ]]

    then

        echo "Der Computer hat gewonnen. Was ein Wunder"

        exit

    fi

    fi

done