
Inhalt

Der Autor	xx
Die Fachgutachterin der amerikanischen Ausgabe	xx
Danksagung	xxi
Einleitung	1
Zielgruppe	2
Der Aufbau dieses Buches	3
Die Codebeispiele	5
Zeilennummern und Einrückungen	5
Lange Codezeilen	6
Python herunterladen und installieren	6
IDLE starten	8
Online Hilfe finden	9
1 Die interaktive Shell	11
Einfaches Rechnen	12
Integer und Fließkommazahlen	13
Ausdrücke	13
Ausdrücke auswerten	14
Syntaxfehler	15
Werte in Variablen speichern	15
Zusammenfassung	19

2 Programme schreiben	21
Stringwerte	22
Stringverkettung	23
Programme im Dateieditor von IDLE schreiben	23
Das Hello-World-Programm schreiben	24
Das Programm speichern	25
Das Programm ausführen	26
So funktioniert das Hello-World-Programm	27
Kommentare	27
Funktionen: Miniprogramme innerhalb von Programmen	28
Das Ende des Programms	29
Variablennamen	30
Zusammenfassung	31
3 Zahlen raten	33
Ein Beispieldurchlauf des Zahlenratespiels	34
Der Quellcode für das Zahlenratespiel	35
Das Modul random importieren	36
Zufallszahlen mit der Funktion random.randint() erzeugen	37
Den Spieler begrüßen	38
Flusssteuerungsanweisungen	39
Code in Schleifen wiederholen	39
Blöcke	39
for-Schleifen	41
Die Vermutung des Spielers abrufen	42
Werte mit den Funktionen int(), float() und str() umwandeln	42
Boolesche Werte	44
Vergleichsoperatoren	45
Bedingungen	45
Experimente mit booleschen Werten, Vergleichsoperatoren und Bedingungen	46
Der Unterschied zwischen = und ==	47
if-Anweisungen	48
Schleifen mit break vorzeitig abbrechen	48
Wenn der Spieler gewonnen hat	48
Wenn der Spieler verloren hat	49
Zusammenfassung	50

4 Ein Kalauerprogramm	53
Ein Beispieldurchlauf von Jokes	54
Der Quellcode für Jokes	54
Funktionsweise des Codes	55
Maskierungszeichen	55
Einfache und doppelte Anführungszeichen	56
Der Schlüsselwortparameter end der Funktion print()	57
Zusammenfassung	58
5 Im Reich der Drachen	59
Spielverlauf von Dragon Realm	59
Ein Beispieldurchlauf von Dragon Realm	60
Das Flussdiagramm für Dragon Realm	60
Quellcode von Dragon Realm	61
Die Module random und time importieren	63
Funktionen	63
def-Anweisungen	63
Funktionen aufrufen	64
Wohin mit Funktionsdefinitionen?	64
Mehrzeilige Strings	65
while-Schleifen	65
Boolesche Operatoren	66
Der Operator and	67
Der Operator or	68
Der Operator not	68
Ausdrücke mit booleschen Operatoren auswerten	69
Rückgabewerte	70
Globaler und lokaler Gültigkeitsbereich	71
Funktionsparameter	72
Das Ergebnis anzeigen	73
Die Höhle mit dem freundlichen Drachen bestimmen	74
Die Hauptschleife des Spiels	75
Die Funktionen aufrufen	75
Neue Runde	76
Zusammenfassung	77

6 Der Debugger	79
Fehlerarten	80
Der Debugger	81
Den Debugger starten	81
Das Programm im Debugger schrittweise durchlaufen	82
Fehler finden	86
Haltepunkte setzen	89
Haltepunkte verwenden	90
Zusammenfassung	92
7 Galgenmännchen: Entwurf mit einem Flussdiagramm	93
Die Regeln für Galgenmännchen	94
Ein Beispieldurchlauf von Galgenmännchen	94
ASCII-Grafik	95
Programme mit Flussdiagrammen entwerfen	96
Das Flussdiagramm zeichnen	97
Verzweigungen in Flussdiagrammen	98
Das Spiel beenden oder neu starten	100
Erneut raten	101
Rückmeldung an den Spieler	102
Zusammenfassung	103
8 Galgenmännchen: Der Code	105
Der Quellcode von Galgenmännchen	106
Das Modul random importieren	109
Konstanten	109
Der Datentyp für Listen	110
Zugriff auf Listenelemente über den Index	110
Listenverkettung	112
Der Operator in	112
Methoden	113
Die Listenmethoden reverse() und append()	113
Die Methode split()	114
Ein Wort aus der Liste auswählen	114
Die Grafik anzeigen	115
Die Funktionen list() und range()	116
Listen- und Stringslices	117
Das zu erratende Wort mit Leerstellen anzeigen	118

Die Rateversuche des Spielers abrufen	120
Die Stringmethoden lower() und upper()	121
Die while-Schleife verlassen	122
elif-Anweisungen	122
Die Eingabe eines gültigen Werts sicherstellen	123
Dem Spieler eine weitere Runde anbieten	124
Die Funktionen im Galgenmännchen-Programm	125
Die Hauptschleife des Spiels	126
Die Funktion displayBoard() aufrufen	126
Den Spieler raten lassen	126
Gehört der Buchstabe zu dem Wort?	127
Hat der Spieler gewonnen?	127
Fehlversuche handhaben	128
Hat der Spieler verloren?	128
Das Spiel beenden oder zurücksetzen	129
Zusammenfassung	130
9 Galgenmännchen: Erweiterungen	131
Mehr Rateversuche hinzufügen	132
Dictionarys	132
Die Größe eines Dictionarys mit len() bestimmen	133
Der Unterschied zwischen Dictionarys und Listen	134
Die Dictionary-Methoden keys() und values()	135
Dictionarys in Galgenmännchen	135
Zufällige Auswahl aus einer Liste	136
Elemente aus Listen entfernen	137
Mehrfachzuweisung	139
Die Wortkategorie ausgeben	140
Zusammenfassung	140
10 Tic-Tac-Toe	143
Ein Beispieldurchlauf von Tic-Tac-Toe	144
Der Quellcode von Tic-Tac-Toe	145
Das Programm entwerfen	149
Das Spielbrett in Form von Daten darstellen	150
Die Strategie der Spiel-KI	150
Das Modul random importieren	152
Das Spielbrett anzeigen	152

Den Spieler zwischen X und O wählen lassen	153
Den ersten Zug auswürfeln	154
Eine Markierung auf dem Spielbrett vornehmen	155
Listenverweise	155
Listenverweise in makeMove()	158
Hat der Spieler gewonnen?	159
Die Spielbrettdaten duplizieren	161
Ist ein Feld belegt?	161
Einen Zug vom Spieler abrufen	162
Kurzschlussauswertung	163
Einen Zug von einer Zugliste auswählen	165
Der Wert None	166
Die KI gestalten	167
Prüfung auf einen Gewinnzug des Computers	167
Prüfung auf einen Gewinnzug des Spielers	168
Ecken, Mitte und Seitenfelder prüfen	169
Ist das Brett voll?	169
Die Hauptschleife des Spiels	170
Den Spielerbuchstaben auswählen und den ersten Zug auslösen	170
Der Zug des Spielers	171
Der Zug des Computers	172
Eine weitere Spielrunde anbieten	173
Zusammenfassung	173
11 Bagels	175
Ein Beispieldurchlauf von Bagels	176
Der Quellcode von Bagels	177
Das Flussdiagramm für Bagels	179
Import von random() und Definition von getSecretNum()	179
Ziffern durcheinanderwürfeln	180
Reihenfolge von Listenelementen mit random.shuffle() ändern	180
Die Geheimzahl aus durcheinandergewürfelten Ziffern zusammenstellen	181
Erweiterte Zuweisungsoperatoren	181
Berechnungen für die Hinweise	182
Die Listenmethode sort()	183

Die Stringmethode <code>join()</code>	184
Besteht der String nur aus Ziffern?	185
Das Spiel starten	185
Stringinterpolation	186
Die Hauptschleife des Spiels	187
Die Vermutung des Spielers abrufen	188
Die Hinweise gewinnen	188
Auf Sieg oder Niederlage prüfen	188
Nach einer weiteren Runde fragen	189
Zusammenfassung	189
12 Kartesische Koordinaten	191
Raster und kartesische Koordinaten	192
Negative Zahlen	194
Das Koordinatensystem eines Computerbildschirms	196
Rechentricks	196
Trick 1: Ein Minuszeichen verschlingt ein Pluszeichen zu seiner Linken	197
Trick 2: Zweimal Minus gibt Plus	197
Trick 3: Zwei Zahlen in einer Addition können den Platz tauschen ..	197
Beträge und die Funktion <code>abs()</code>	198
Zusammenfassung	199
13 Sonar-Schatzsuche	201
Ein Beispieldurchlauf der Sonar-Schatzsuche	203
Der Quellcode für die Sonar-Schatzsuche	205
Entwurf des Programms	210
Die Module <code>random</code> , <code>sys</code> und <code>math</code> importieren	211
Ein neues Spielbrett erstellen	211
Das Spielbrett zeichnen	213
Die x-Koordinaten am oberen Rand des Spielbretts zeichnen	213
Das Meer zeichnen	214
Eine Meeresreihe ausgeben	215
Die x-Koordinaten am unteren Rand ausgeben	216
Die zufällig verteilen Schatztruhen erstellen	216
Die Gültigkeit eines Zugs bestimmen	217

Einen Zug auf dem Spielbrett machen	217
Die nächstgelegene Schatztruhe finden	218
Werte mit der Listenmethode <code>remove()</code> entfernen	220
Den Zug des Spielers abrufen	222
Die Spielanleitung ausgeben	224
Die Hauptschleife des Spiels	224
Den Spielzustand anzeigen	225
Den Zug des Spielers verarbeiten	226
Eine Schatztruhe finden	226
Hat der Spieler gewonnen?	227
Hat der Spieler verloren?	227
Das Programm mit der Funktion <code>sys.exit()</code> beenden	228
Zusammenfassung	228
14 Die Caesar-Chiffre	231
Kryptografie und Verschlüsselung	232
Funktionsweise der Caesar-Chiffre	233
Ein Beispieldurchlauf der Caesar-Chiffre	234
Der Quellcode für die Caesar-Chiffre	235
Die maximale Schlüssellänge festlegen	236
Auswahl zwischen Ver- und Entschlüsselung	237
Die Nachricht vom Benutzer abrufen	237
Den Schlüssel vom Benutzer abrufen	237
Die Nachricht verschlüsseln oder entschlüsseln	238
Strings mit der Methode <code>find()</code> finden	239
Die einzelnen Buchstaben ver- bzw. entschlüsseln	240
Das Programm starten	241
Brute-Force-Entschlüsselung	241
Den Brute-Force-Modus hinzufügen	242
Zusammenfassung	243
15 Reversegam	245
Die Spielregeln von Reversegam	246
Ein Beispieldurchlauf von Reversegam	249
Der Quellcode von Reversegam	251
Module importieren und Konstanten einrichten	257
Die Datenstruktur für das Spielbrett	257

Die Datenstruktur für das Spielbrett auf dem Bildschirm ausgeben . . .	258
Eine neue Datenstruktur für das Brett erstellen	259
Züge auf Gültigkeit prüfen	259
Die acht Richtungen prüfen	260
Prüfen, ob Steine gedreht werden müssen	262
Prüfen auf gültige Koordinaten	263
Eine Liste mit allen gültigen Zügen bekommen	263
Die Funktion bool() aufrufen	264
Die Punktzahl des Spielbretts ermitteln	265
Die Steinwahl des Spielers abfragen	266
Entscheiden, wer beginnt	266
Einen Spielstein auf dem Brett platzieren	267
Die Datenstruktur des Spielbretts kopieren	268
Bestimmen, was eine Ecke ist	268
Den Spielzug abfragen	268
Der Zug des Computers	270
Strategische Züge in Ecken	271
Eine Liste der besten Züge erstellen	271
Die Punkte auf dem Bildschirm ausgeben	273
Das Spiel starten	273
Ein Patt erkennen	274
Der Spieler ist am Zug	274
Der Spielzug des Computers	276
Die Spielschleife	277
Fragen, ob der Spieler erneut spielen möchte	278
Zusammenfassung	278
16 Reversegam-KI-Simulation	279
Der Computer spielt gegen sich selbst	280
Ein Beispieldurchlauf von Simulation 1	281
Der Quellcode für Simulation 1	281
Die Spielereingaben entfernen und einen Computerspieler hinzufügen	283
Den Computer mehrfach gegen sich selbst spielen lassen	284
Ein Beispieldurchlauf von Simulation 2	284
Quellcode für Simulation 2	284
Den Überblick über mehrere Spiele behalten	285

Die Funktionsaufrufe print() auskommentieren	286
Die KI mit Prozentsätzen bewerten	287
Die verschiedenen KI-Algorithmen vergleichen	289
Quellcode für Simulation 3	289
Wie die KI in Simulation 3 funktioniert	291
Die KI miteinander vergleichen	294
Zusammenfassung	296
17 Grafik einsetzen	299
pygame installieren	300
Hello World in pygame	301
Beispieldurchlauf des pygame Hello World	301
Quellcode für pygame Hello World	302
Das Modul pygame importieren	303
pygame initialisieren	304
Das pygame-Fenster einrichten	304
Tupel	305
Surface-Objekte	305
Farb-Variablen einrichten	306
Im pygame-Fenster Text ausgeben	307
Text mit Schriftarten formatieren	307
Ein Font-Objekt ausgeben	308
Die Textposition mit Rect-Attributen festlegen	309
Ein Surface-Objekt mit Farbe füllen	311
Die Zeichenfunktionen von pygame	311
Ein Polygon zeichnen	312
Eine Linie zeichnen	313
Einen Kreis zeichnen	313
Eine Ellipse zeichnen	314
Ein Rechteck zeichnen	314
Pixel färben	315
Die blit()-Methode für Surface-Objekte	316
Das Surface-Objekt auf dem Bildschirm ausgeben	316
Ereignisse und die Game-Schleife	316
Ereignis-Objekte erhalten	317
Das Programm beenden	318
Zusammenfassung	318

18 Grafiken animieren	319
Beispieldurchlauf des Animationsprogramms	320
Der Quellcode für das Animationsprogramm	320
Kästen bewegen und abprallen lassen	322
Die Konstanten einrichten	323
Konstanten für die Richtung	324
Konstanten für die Farbe	325
Datenstrukturen für die Kästen einrichten	325
Die Game-Schleife	326
Wenn der Spieler aufhören möchte	326
Die Kästen bewegen	327
Einen Kasten abprallen lassen	328
Kästen an ihren neuen Positionen im Fenster zeichnen	329
Das Fenster auf den Bildschirm zeichnen	330
Zusammenfassung	330
19 Kollisionserkennung	331
Beispieldurchlauf der Kollisionserkennung	332
Der Quellcode für die Kollisionserkennung	333
Die Module importieren	335
Das Programm mit einem Zeitgeber steuern	335
Das Fenster und die Datenstrukturen einrichten	336
Variablen zur Bewegungsverfolgung einrichten	337
Ereignisse verarbeiten	338
Das KEYDOWN-Ereignis	339
Das Ereignis KEYUP	342
Den Spieler teleportieren	342
Neue Nahrungsquadrate hinzufügen	343
Den Spieler im Fenster bewegen	344
Den Spieler im Fenster zeichnen	344
Auf Kollisionen prüfen	345
Die Nahrungsquadrate im Fenster zeichnen	346
Zusammenfassung	346

20 Toneffekte und Bilder	347
Bilder mit Sprites hinzufügen	348
Ton- und Bilddateien	349
Beispieldurchlauf des Sprites- und Sound-Programms	349
Der Code für das Sprites- und Sounds-Programm	350
Das Fenster und die Datenstruktur einrichten	353
Ein Sprite hinzufügen	353
Die Größe eines Sprites verändern	354
Die Musik und die Toneffekte einrichten	354
Sound-Dateien hinzufügen	354
Den Ton ein- und ausschalten	355
Den Spieler im Fenster zeichnen	355
Auf Kollisionen prüfen	356
Die Kirschen im Fenster zeichnen	356
Zusammenfassung	357
21 Dodger mit Ton und Grafik	359
Wiederholung: Die grundlegenden Datentypen in pygame	360
Ein Beispieldurchlauf von Dodger	361
Der Quellcode von Dodger	361
Die Module importieren	366
Die Konstanten einrichten	366
Funktionen definieren	367
Das Spiel beenden und unterbrechen	367
Kollisionen mit Gegnern feststellen	368
Text in das Fenster zeichnen	369
pygame initialisieren und das Fenster einrichten	370
Schrift-, Klang- und Bildobjekte einrichten	371
Den Startbildschirm anzeigen	372
Das Spiel starten	373
Die Spielschleife	374
Tastaturereignisse	375
Mausbewegungen	376
Neue Gegner hinzufügen	377
Die Figur des Spielers und die Gegner bewegen	378
Die Cheats	379

Gegner entfernen	380
Das Fenster zeichnen	380
Den Punktestand ausgeben	381
Die Spielerfigur und die Gegner zeichnen	381
Auf Kollisionen prüfen	382
Game Over	383
Das Spiel abwandeln	383
Zusammenfassung	384
Stichwortverzeichnis	385