Python Kurs, Martin Reiche 2024
Version vom 7.5.2024

Start-Infos für das Projekt „Sokoban“

Phase 1

Informiere dich über das Spiel in der Wikipedia und in YouTube. Welcher Logik folgt es?

Unser Modell des Spielplans ist eine Matrix, in der jedes Element entweder leer ist ( ) oder vom Spieler (@), einer Kiste ($) oder einem Wandelement (#) belegt wird. Sie ist in Textform codiert. Ein erster Plan zum Testen lautet:

```
plan = '''
#######
#    .#
# #$###
#@   ##
#######'''
```

Das hier einzige Ziel ist mit einem Punkt (.) markiert.

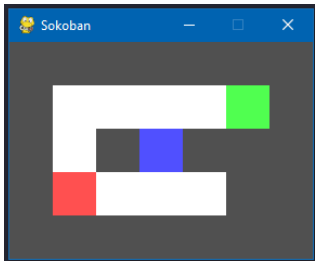
Schreibe ein Konsolenprogramm, das nach jeder Bewegung des Spielers den aktualisierten Plan ausgibt. Es endet, wenn alle Kisten im Ziel liegen. Die Steuerung des Spielers erfolgt mit den Tasten des Zahlenblocks rechts auf der Tastatur (8 = hoch, 2 = runter, 4 = links, 6 = rechts), gefolgt von der Eingabe-Taste.

Tipps:

- Die Matrix *store* des Plans ist eine geschachtelte Liste.
- Splitte den Plan am Zeilenumbruch zur Erzeugung der Matrix!
- Für den Startpunkt des Spielers im obigen Plan gilt: store[3][1] == ‘@‘.
- Benutze die join-Funktion zur Umwandlung einer Matrix-Zeile in einen String! Beispiel:
  a = ['1', '2', '3']
  ''.join(a) liefert '123'

Man benötigt etwa 70 Codezeilen. Ersetzt man den Testplan durch den Plan aus http://tuepy.wikidot.com/projects, so kann man das gesamte Spiel schon jetzt spielen! Es fehlt „nur noch“ eine ansprechende Grafik.

Phase 2



Nachdem du die Logik des Spiels programmiert hast, kannst du Pygame zur grafischen Darstellung benutzen. Es soll für den Testplan ein Bild wie das rechts dargestellte resultieren: Spieler rot, Kiste blau, Ziel grün, Wände grau, Gänge weiß.

Gehe wie folgt vor:

1. Installiere Pygame in Thonny über das Menü Extras > Verwalte Pakete. Gebe *Pygame* in das Suchfeld oben ein und drücke auf „Suche“. Wähle „pygame - Python Game Development“ und installiere!
2. Folge dem Video in https://youtu.be/hxYT5fwEK1A?si=3gpaH1uPSxbpAxrM&t=225 ab dem gewählten Zeitpunkt für eine kurze Einführung in Pygame.
3. Lade die beiliegende Datei *SokobanStarter.py* und führe sie aus. Beobachte die Ausgabe auf der Konsole!
4. Implementiere die Funktion *init_view* zu Darstellung des Spielplans. Benutze die Funktionen *pygame.draw.rect(screen, color, …)*, um die Kästchen zu malen. Siehe auch https://www.pygame.org/docs/ref/draw.html#pygame.draw.rect und https://www.pygame.org/docs/ref/rect.html#pygame.Rect.
5. Implementiere die *Funktion update_game.* Das wird der größte Block, weil hier die gesamte Logik des Spiels versammelt ist. Gegebenenfalls teile die Funktionalität dort auf in weitere Unterfunktionen. Wenn du die Bewegung des Lageristen und das Schieben einer Kiste flüssig gestalten willst, rufe *pygame.display.update()* und *pygame.time.delay(30)* in einer Schleife.
6. Der *event.key* als Argument von *update_game* enthält Codes, die hier https://www.pygame.org/docs/ref/key.html#key-constants-label verzeichnet sind und gegen die ihr prüfen müsst. Wenn du den Spieler mit den Pfeiltasten führen willst, sind das die Codes K_UP etc.

Phase 3

In dieser Phase kommen dann die Grafiken und die Animation des Lageristen hinzu.

Ganz zum Schluss kannst du z.B. noch Levels und Bestenlisten hinzufügen.