

Übung 5

1. Stellen Sie die Fehlerbehandlung in der Klasse auf das C++ Exception Handling mit Standard C++ Exceptions um (Z.B. `std::bad_alloc` für eine nicht erfolgreiche Speichieranforderung)
2. Erweitern Sie die Klasse um den Operator `[]` um über einen Index lesend auf einzelne Zeichen zugreifen zu können (`const char operator[] (unsigned) const`). Verwenden Sie das Exception Handling um im Falle eines ungültigen Index eine Exception zu „werfen“ (`std::out_of_range`).

Zusammen mit der bereitgestellten „main.cpp“ Datei sollte folgende Ausgabe erfolgen:

```
Ausgewaehltes Wort: >Beispiel<
Selektieren Sie einen Buchstaben [1-8):
9
Falsche Auswahl - noch einmal versuchen ...
Ausgewaehltes Wort: >Beispiel<
Selektieren Sie einen Buchstaben [1-8):
2
Selektierter Buchstabe: >e<
```



```
elif operation == "MIRROR_X":
    mirror_mod.use_x = True
    mirror_mod.use_y = False
    mirror_mod.use_z = False
elif operation == "MIRROR_Y":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

#selection at the end -add back the deselected mirror modifier object
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active ob
#mirror_ob.select = 0
#new = bpy.context.selected_objects[0]
#new.data.objects[mirror_ob.name].select = 0
```