

Praktikum Aufgabe 3 – Tastaturtreiber

Punktzahl gesamt: 100 Punkte

Abgabe der Lösungen bis spätestens 18.02.21 10:00 Uhr

Aufgabe 1 – Abfangen eines Tastendrucks

30 Punkte

Zuerst sollen Sie die Methode **key_hit** in der Klasse **Keyboard_Controller** in der Datei **machine/keyctrl.cc** implementieren. Den **key** erhalten Sie über den **data_port**, indem Sie auf diesem **inb()** aufrufen. Weiterhin sollten Sie testen, dass Sie keine PS2 Tastatur benutzen. Dies können Sie über **auxb** überprüfen.

Danach sollten Sie den Tastencode decodieren. (Siehe: **machine/keydecoder**).

Zusätzlich sollen Sie die Methode **prologue** der **device/keyboard** implementieren. Den entsprechenden **key** bekommen sie von **key_hit**. Die Methode soll zunächst überprüfen ob der **currentkey** NICHT mehr gültig ist. Ist dies der Fall, wurde der Tastendruck bereits verarbeitet und der **currentkey** kann auf den aktuellen Key (**key**) gesetzt werden. Ist dies erfolgreich, soll **true** zurückgeben werden. Sonst wird **false** zurückgegeben.

Weiterhin soll in der Methode bei der Tastenkombination **strg + alt + entf** das Betriebssystem neu gestartet werden. Es kann vorkommen, dass die Tastenkombination bei manchen Systemen nicht korrekt an OOSTuBS übergeben wird. Falls dies zutreffen sollte, verwenden sie eine eigene Tastenkombination und kommentieren Sie dies entsprechend.

Aufgabe 2a – Sondertasten

20 Punkte

Nun müssen Sie, innerhalb der **machine/key.cc** die Methoden, die ein **TODO** enthalten, implementieren. Dazu haben Sie mehrere Beispiele gegeben.

Bei den Methoden, **alt()** und **ctrl()** ist zu beachten, dass es für unser System, keinen Unterschied macht, ob die jeweilige linke oder rechte Taste gedrückt wurde.

Aufgabe 2b – Eingabemasken

20 Punkte

Als nächstes wird von Ihnen gefordert in der **machine/keydecoder.cc**, die Masken (mit den **TODOs** darüber) für die einzelnen Tasten korrekt zu ergänzen. Dabei müssen Sie alle # durch entweder die korrekten Zeichen in Hochkommata oder den ASCII Codes als Zahlen OHNE Hochkommata ergänzen.

Beispiel: Das # wäre also '#' und β wäre 225

Die Nullen stehen für leere bzw ungültige Tasten diese müssen Sie nicht beachten. OOSTuBS und gemu verwenden die erweiterte ASCII-Tabelle von IBM. Ihr findet die richtige Tabelle unter

<https://www.ascii-codes.com/>

Aufgabe 3 – Ausgabe auf der Konsole

20 Punkte

In dieser Aufgabe beschäftigen Sie sich mit dem Konzept einer Tasteneingabe. Dafür müssen Sie innerhalb der **device/keyboard.cc** die Methoden **epilog** und **getkey** implementieren. In der **epilog**-Methode soll die Verfügbarkeit der Aufnahme eines Tastendruckes implementiert werden. Dazu ist es sinnvoll zu verstehen, dass es sich bei dem Bearbeiten eines Tastendruckes um einen kritischen Abschnitt handelt. Innerhalb der **getkey**-Methode soll nun die aktuelle Taste in einen **key** gespeichert werden, danach soll die aktuelle Taste ungültig gemacht werden und der **key** zurück gegeben werden. Tipp für den kritischen Abschnitt: **meeting/semaphore.h**

Aufgabe 4 – Testen, Testen, Testen

10 Punkte

Zum Schluss sollen Sie nun noch eine simple Anwendung schreiben, die ihren Keyboardtreiber testet. Dieser soll in der **user/app/kappl.cc** implementiert werden. Testen Sie die Tasten:

- **s**
- **T**
- **)**
- **{**
- **\n** (Enter)
- **\b** (Backspace)

Der Test soll dabei durch einen einfachen Vergleich mit dem richtigen Zeichen passieren (**testC**-Methode).