

1) Zeiten schätzen

Es soll ein Spiel entwickelt werden, bei dem der Spieler eine vorgegebene Zeit von 60 Sekunden möglichst gut schätzen soll. Es gibt 600 Punkte und für jede Zehntelsekunde, die der Spieler daneben liegt ein Punkt Abzug.

Beispiel: Drückt der Spieler nach 62,3 Sekunden eine Taste hat er 577 Punkte (600-23).

Hinweise zur Implementierung

- Versuche möglichst frühzeitig das Programm auf der Megacard zu testen
- Verkürze eventuell die 60 Sekunden während der Implementierung, um nicht jedesmal 60 Sekunden warten zu müssen. Beim abgegebenen Projekt müssen die 60 Sekunden aber stimmen!
- Kontrolliere bei Fertigstellung der Implementierung die Übereinstimmung mit der Spezifikation
- Lege bei Fertigstellung einer Teilaufgabe eine Sicherungskopie an, um eventuell darauf zurückzukehren
- Abgabe erfolgt durch Speicherung auf das R: Laufwerk oder auf einem USB Stick
- Die Abgaben werden nicht beurteilt wenn unmittelbaren Ähnlichkeiten zwischen zwei Abgaben gefunden werden

Das Spiel soll mittels der folgenden vier Zustände realisiert werden:

Zustand Begrüßung

S	T	A	R	T															

- Der Bildschirm zeigt „START“
- Bei einem Tastendruck (S0-S3) soll in den Zustand *Vorbereitung* gewechselt werden

Zustand Vorbereitung

W	A	I	T	!															

- Die erste Zeile zeigt „WAIT“
- Wenn drei Sekunden abgelaufen sind soll in den Zustand *Spielen* gewechselt werden

Zustand Spielen

G	O	!																	

- Die erste Zeile zeigt „GO!“
- Der Spieler drückt nach seinen geschätzten 60 Sekunden auf die Taste S0 um in den Zustand *Spiel Ende* zu kommen

Zustand Spiel Ende

2	7	3																	

- Die erste Zeile zeigt den erzielten Punktestand (600 Punkte minus den Zehntel Sekunden Abweichung)
- Die Abzüge gelten für eine Zeit über und unter 60 Sekunden
- Es kann auch ein negativer Punktestand erreicht werden
- Der angezeigte Punktestand ist linksbündig
- Durch Drücken der Taste S3 soll die Begrüßung angezeigt werden

1.a) Funktionsfähigkeit

12 Punkte

Bei der Funktionsfähigkeit wird für jeden vollständig implementierten und funktionierenden Zustand drei Punkte vergeben. Die Angabe dient als Spezifikation und soll genau eingehalten werden. Jede Nichteinhaltung führt zu Punkteabzügen.

1.b) Strukturierung

6 Punkte

- Bewertet den Aufbau, Namenswahl der Funktionen und Variablen
- Wähle sprechende Namen, wenn möglich in Englisch
- Achte auf eine möglichst sinnvolle Einrückung des Codes
- Jeder Codeteil muss durch Kommentare die Funktion zusammenfassen

2) Erweiterungen

Die Erweiterungen können unabhängig voneinander implementiert werden. Wähle die Reihenfolge nach eigenem Ermessen.

2.a) Rückkehr zum Zustand *Begrüßung* 1 Punkt

- In den Zuständen *Vorbereitung* und *Spielen* soll durch Drücken der Taste S3 zum Zustand *Begrüßung* gewechselt werden
- Der erzielte Punktestand spielt keine Rolle.

2.b) Anzeigen des maximalen Punktestandes 4 Punkte

- Der maximal erreichte Punktestand soll gespeichert und angezeigt werden
- Nach dem Einschalten der Spannungsversorgung soll der maximale Punktestand 0 sein
- Die Anzeige beim Zustand *Begrüßung* wird dazu verändert (hier mit 387 maximal erreichten Punkten):

S	T	A	R	T				
M	A	X		3	8	7		

2.c) Zwei Spieler Erweiterung 6 Punkte

- Das Spiel soll für zwei Spieler erweitert werden
- Der zweite Spieler drückt die Taste S1 zum Spielen
- Zum Zustand *Spiel Ende* wird gewechselt, nachdem beide Spieler ihre Taste gedrückt haben
- Bei *Spiel Ende* werden die beiden Punktestände angezeigt (oben Spieler eins, unten Spieler zwei)

3	8	7						
-	5	1	6					

- Für den maximalen Punktestand sollen beide Spieler berücksichtigt werden

2.d) *Abbruch nach 150 Sekunden* 2 Punkt

- Wenn nach 150 Sekunden ein Spieler noch keine Taste gedrückt hat wechselt das Spiel automatisch in den Zustand *Spiel Ende*
- Der Spieler, der keine Taste gedrückt hat, erhält -900 Punkte