
Concurrent Programming<http://www-pu.informatik.uni-tuebingen.de/cp-2006>

Blatt 7

Abgabe: 27.6.2006

1. [4 Punkte] Schreibe ein triviales GUI-Programm mit dem Toy-Window-System: Es soll ein Frame enthalten, in dem sich zwei Buttons mit beliebigen Beschriftungen befinden. Wird ein Button gedrückt, soll eine (je nach Button unterschiedliche) Nachricht ausgedruckt werden.
2. [13+13 Punkte] In dieser Aufgabe geht es darum, Widgets für das Toy-Window-System zu programmieren. Es werden fünf Widgets vorgestellt und Aufgabe deines Teams ist es, zwei Widgets auszuwählen und zu implementieren.

Da alle Widgets in den Aufgaben des nächsten Übungsblattes verwendet werden, muss sichergestellt werden, dass von jedem Widget mindestens eine Implementierung vorliegt. Es ist darum unbedingt erforderlich, dass ihr eure Wünsche möglichst früh per E-Mail bei Eric (knauel@informatik.uni-tuebingen.de) anmeldet. Sollte es erforderlich sein, wird Eric dein Team mit der Implementierung eines anderen Widgets beauftragen.

Die folgenden Widgets stehen zur Auswahl:

- (a) *Slider* Ein Slider ist ein Schieberegler, der eine Zahl aus einem festen Intervall repräsentiert — wie z. B. ein Lautstärkereglers. Schreibe dafür eine Prozedur `make-slider`, die als Argumente einen asynchronen Kanal und ein Widget-Environment akzeptiert. `Make-slider` soll den Slider aufbauen und das Widget-Environment zurückliefern. Der Slider sendet die aktuelle Einstellung des Reglers, wann immer sich die Position ändert, in den asynchronen Kanal. Die Einstellung ist eine Zahl zwischen 0.0 und 1.0, entsprechend der Position.
- (b) *Eingabefeld* Ein Eingabefeld nimmt die Tastatureingaben eines Benutzer entgegen, wenn der Maus-Cursor über dem Eingabefeld steht. Ein Eingabefeld wird durch die Funktion `make-text-field` erzeugt, die einen voreingestellten Text, eine Breite, einen asynchronen `read-channel`, einen synchronen `set-channel` als Argumente nimmt. Hat der Benutzer den Text verändert, wird der aktuelle Inhalt in den `read-channel` geschrieben. Um einen neuen Text zu setzen, sendet der Benutzer des Widgets einen String in den `write-channel`. Füllt die Eingabe des Benutzers die gesamte Breite des Widgets, akzeptiert das Widget nur noch die Backspace-Taste, um das letzte Zeichen zu löschen.
- (c) *Stepper* Ein Stepper-Widget ist dem Zahlenschloß eines Fahrrads sehr ähnlich. Es dient dazu eine ganze Zahl einzustellen und besteht aus drei Teilen: Einem Textfeld, das die gerade eingestellte Zahl anzeigt und zwei Knöpfen mit denen man zur jeweils nächsten bzw. vorhergehenden Zahl gelangt. Ein Stepper wird durch `make-stepper` erzeugt und nimmt zwei ganze Zahlen (die Grenzen des Intervalls), einen asynchronen Kanal (enthält die neu eingestellte Zahl, wenn sich diese geändert hat) und einen synchronen Kanal (zum Setzen des Zählers) als Argumente.
- (d) *Radio Button Group* Eine Radio-Button-Group dient der Auswahl einer Option und einer festen Anzahl von Alternativen. Dazu ordnet dieses Widget alle Optionen, die durch einen kurzen Text beschrieben sind, horizontal oder vertikal an. Links des Textes befindet sich ein kleines Feld, auf das der Benutzer klicken kann, um diese Option auszuwählen. Es kann immer nur eine Option auf einmal ausgewählt sein. Der Konstruktor `make-radio-buttons` nimmt als Argumente eine Liste mit den Optionen, einen asynchronen Kanal, auf den Nachrichten ankommen, wenn ein neue Option gewählt wurde und einen synchronen Kanal über den die momentan gewählte Option gesetzt wird.

Die Liste der Optionen ist eine Liste von Paaren, deren erstes Feld jeweils den Text enthält und deren zweites Feld den Wert enthält, der gesendet wird, wenn diese Option ausgewählt wurde.

- (e) *Popup Menu* Ein Popup-Menü bietet eine Auswahl zwischen unterschiedlichen Menü-Punkten. Popup-Menüs werden durch die Funktion `popup-menu` erzeugt. Argumente für diese Funktion sind ein Widget-Environment, die Koordinaten an denen das Menü dargestellt werden soll und eine Liste der Menü-Einträge. `popup-menu` zeigt das Menü an und kehrt zurück, wenn ein Menüpunkt ausgewählt wurde. Die Menü-Einträge werden `popup-menu` als Liste von Paaren übergeben. Das erste Feld eines Paares ist dabei der String, der als Eintrag angezeigt werden soll, der zweite Teil der Wert, den `popup-menu` zurückgeben soll, wenn dieser Eintrag ausgewählt wurde.

Bedenke, dass der Code für deine Widgets eventuell schon nächste Woche von deinen Kommilitonen verwendet wird! Zu einer Widget-Implementierung gehören deshalb eine kurze Dokumentation, die die Verwendung der zur Verfügung gestellten Funktionen erläutert sowie ein Beispielprogramm, das das Widget verwendet.

Hinweis: Die erwähnten Konstruktoren brauchen neben den beschriebenen Argumenten noch weitere Argumente (z. B. für das Widget-Environment).