

## Projektbeschreibung:

In der modernen Arztpraxis melden sich eintreffende Patienten selbst mit Nachname, Vorname und Geburtsdatum am Terminal an. Sie erhalten eine Laufnummer – morgens beginnend mit 1 – angezeigt und die Information, wie viele Patienten noch vor ihnen angemeldet sind, aber noch nicht behandelt wurden. So können sie selbst entscheiden, ob sie zuvor z. B. noch Einkäufe erledigen wollen.

Der Arzt kann im Terminal des Behandlungszimmers jeweils den Namen des nächsten Patienten sehen und aus dem Wartezimmer (ggf. über ein Display dort) aufrufen. Zusätzlich erkennt er, wie lange dieser Patient seit seiner Anmeldung warten musste. Diese Wartezeit soll auch für die Tages-Endauswertung abrufbar sein. Auch soll ablesbar sein, wie viele weitere Patienten noch warten.

## Aufgabenbeschreibung:

Alle nachfolgenden Aufgabenteile sollten anfangs nur mit Papier und Bleistift erarbeitet werden! Später können dann Programmier-Werkzeuge (ArgoUML, StruktEdit, JavaEditor) eingesetzt werden.

- a) Entwerfen Sie für dieses System ein UML-Klassendiagramm, das u. a. die Klassen Praxis, Behandlung und Patient enthalten sollte. Denken Sie auch über abstrakte Klassenbildung und mögliche Spezialisierungen nach.
- b) Erweitern Sie dieses Diagramm um die Klassen GuiAnmeldung, GuiBehandlung und ggf. GuiWartezimmer. Entwerfen Sie jeweils ein Benutzer-Interface (Formular) für die Klassen GuiAnmeldung und GuiBehandlung.
- c) Formulieren Sie als Grobalgorithmus oder in Form eines Struktogrammes die Methode bt\_AnmeldenAction().
- d) Implementieren Sie in Java die Methode bt\_AnmeldenAction() und den/die Patienten-Konstruktor(en). Hinweis: Eine statische (also überall verfügbare) Methode getDateZeit() sei in der Praxis-Klasse verfügbar. Sie gibt die aktuelle Datum-Zeitangabe als Zahl der Form JJJJMMTThhmm zurück, also z. B. 200611271135. Diese Methode muss nicht implementiert werden.

E:\Java\Projekte\Arztpraxis\Arztpraxis-OOA-OOP.doc

## Projektbeschreibung:

In der modernen Arztpraxis melden sich eintreffende Patienten selbst mit Nachname, Vorname und Geburtsdatum am Terminal an. Sie erhalten eine Laufnummer – morgens beginnend mit 1 – angezeigt und die Information, wie viele Patienten noch vor ihnen angemeldet sind, aber noch nicht behandelt wurden. So können sie selbst entscheiden, ob sie zuvor z. B. noch Einkäufe erledigen wollen.

Der Arzt kann im Terminal des Behandlungszimmers jeweils den Namen des nächsten Patienten sehen und aus dem Wartezimmer (ggf. über ein Display dort) aufrufen. Zusätzlich erkennt er, wie lange dieser Patient seit seiner Anmeldung warten musste. Diese Wartezeit soll auch für die Tages-Endauswertung abrufbar sein. Auch soll ablesbar sein, wie viele weitere Patienten noch warten.

## Aufgabenbeschreibung:

Alle nachfolgenden Aufgabenteile sollten anfangs nur mit Papier und Bleistift erarbeitet werden! Später können dann Programmier-Werkzeuge (ArgoUML, StruktEdit, JavaEditor) eingesetzt werden.

- a) Entwerfen Sie für dieses System ein UML-Klassendiagramm, das u. a. die Klassen Praxis, Behandlung und Patient enthalten sollte. Denken Sie auch über abstrakte Klassenbildung und mögliche Spezialisierungen nach.
- b) Erweitern Sie dieses Diagramm um die Klassen GuiAnmeldung, GuiBehandlung und ggf. GuiWartezimmer. Entwerfen Sie jeweils ein Benutzer-Interface (Formular) für die Klassen GuiAnmeldung und GuiBehandlung.
- c) Formulieren Sie als Grobalgorithmus oder in Form eines Struktogrammes die Methode bt\_AnmeldenAction().
- d) Implementieren Sie in Java die Methode bt\_AnmeldenAction() und den/die Patienten-Konstruktor(en). Hinweis: Eine statische (also überall verfügbare) Methode getDateZeit() sei in der Praxis-Klasse verfügbar. Sie gibt die aktuelle Datum-Zeitangabe als Zahl der Form JJJJMMTThhmm zurück, also z. B. 200611271135. Diese Methode muss nicht implementiert werden.

E:\Java\Projekte\Arztpraxis\Arztpraxis-OOA-OOP.doc