

Übungsblatt 14

Ausgabe: 4.2.

Abgabe: Mi., 11.2., 9:45 Uhr, eClaus.informatik.uni-stuttgart.de

Abgabe erfolgt ausschließlich elektronisch über eClaus.informatik.uni-stuttgart.de – versuchen Sie nach Möglichkeit die Abgabe nicht in der letzten Minute zu machen!

Von jedem Aufgabenblatt werden maximal 20 Punkte auf den Schein angerechnet.

+++WICHTIG+++ Als Übungstermin für die Mechatroniker-Informatik-II-Übungsgruppe ist im Sommer Dienstag 9h45-11h15 vorgesehen. Falls dies bei Ihnen mit anderen Veranstaltungen kollidiert, sagen Sie bitte frühstmöglich Bescheid! +++WICHTIG+++

1. (2+6+1 Punkte, mittel–schwer) **Ein kleiner Taschenrechner:** Diese Aufgabe verzahnt die verschiedenen Gebiete, die wir in der Informatik bisher kennengelernt haben.

Gegeben sei folgende Grammatik, die einfache arithmetische Ausdrücke mit ganzen Zahlen erzeugt: $G = (V, \Sigma, P, S)$, mit $V = \{S, A, M, Z\}$ und $P = \{S \rightarrow S + A, S \rightarrow S - A, S \rightarrow A, A \rightarrow A * M, A \rightarrow A/M, A \rightarrow M, M \rightarrow (S), M \rightarrow Z, Z \rightarrow 0Z, \dots, Z \rightarrow 9Z, Z \rightarrow 0, \dots, Z \rightarrow 9\}$. Die Idee hinter dieser Modellbildung: Ein arithmetischer Ausdruck (unter Berücksichtigung von Punkt- vor Strichrechnung) ist eine Folge von Summanden, jeder dieser Summanden ist ein Produkt von Faktoren und jeder Faktor ist entweder ein Klammerausdruck oder eine Zahl.

- Begründen Sie, warum diese Grammatik eindeutig ist (bitte als Kommentarzeilen mit einfügen). Beschreiben Sie kurz, welche Vorteile diese Eindeutigkeit haben könnte?
 - Schreiben Sie eine Funktion, die einen String einliest (`Get_Line(string, integer)` liest eine Zeile ein und gibt in der Variablen `integer` zurück, wieviel Zeichen tatsächlich eingelesen wurden) und diesen dann auswertet. Versuchen Sie dabei die obigen Regeln der Grammatik zu benutzen (schreiben Sie Funktionen `S`, `A`, `M` und `Z`, die jeweils einen Integer-Wert berechnen – die Funktion `S` sucht z.B. das letzte `+` oder `-` im übergebenen String und gibt als Ergebnis die Summe von `S` (Teilstring vor dem `+/-` als Parameter) und `A` (Teilstring nach dem `+/-` als Parameter) zurück).
Hinweis: Falls Sie die obige Anleitung eher verwirrt, überlegen Sie selbstständig, wie Sie aus dem Eingabestring die Zeichen in der richtigen Reihenfolge entnehmen und auswerten können.
Hinweis 2: Fehlerhafte Eingaben müssen nicht erkannt werden – es reicht, wenn Ihre Funktion korrekt eingegebene arithmetische Ausdrücke korrekt auswerten kann.
 - Beschreiben Sie, in welchem Zusammenhang Ihr Vorgehen bei der Lösung mit den behandelten Baumdurchlaufalgorithmen (Pre-, In-, Postorder) steht (bitte als Kommentarzeilen mit einfügen).
2. (10 Punkte, mittel–schwer) **Labyrinth erstellen:** Mittels der ASCII-Zeichen `|` (senkrechter Strich) und `-` (Minus-Zeichen) lassen sich vertikale und horizontale Linien auf dem Bildschirm zeichnen, wenn man die Ausgabe entsprechend mit Leerzeichen anreichert und einen Font mit festem Zeichenabstand einstellt (ganz wichtig). An den Stellen, wo sich die Richtung einer Linie ändert oder eine Abzweigung stattfindet, soll ein `+` (Pluszeichen) erscheinen.

Schreiben Sie ein Programm, das die Zeilen- und Spaltenzahl eines rechteckigen Bereiches einliest und dann ein zufälliges Labyrinth berechnet und ausgibt. Ihr Labyrinth sollte eine Umrandung mit einem Eingang (Leerstelle) links oben und einen Ausgang rechts unten haben. Gänge sind nur dann "begehrbar", wenn zwischen den "Wänden" mindestens ein Leerzeichen zum Durchlaufen existiert. Es sollen keine Gänge mit mehr als 1 Leerzeichen vorkommen. Wände sollen stets die Breite 1 haben (d.h., in der Ausgabe hat das Labyrinth immer ungerade Zeilen- und Spaltenanzahlen).

Natürlich soll es stets einen Weg von links oben nach rechts unten geben. Schaffen Sie es, die Labyrinth so entwerfen zu lassen, dass es immer nur genau einen Weg gibt?

Hinweis: Manchmal ist es hilfreich, das Problem andersherum aufzuziehen. Versuchen Sie z.B. zuerst die Wege zu erstellen und die Wände danach um die Wege herum zu bauen.

```

+++++
+-----+ | +-- +-- +-----+-- | | +-+ +-+ +-----+-----+
+
+-- +-+---+ +-+ --+ ---+-----+ +-----+ +-+ | +-+--- +-----+
+
+ -+ +-----+---+ +-+ | -+---+-----+ +-+ +-+ | -+ -+-----+ +
+
+ ---+---+ +-+ | -----+ +-+ | -----+-----+ +-+ | | |
+
+-----+ | | +-----+-----+ +-+ | | +-+ | +-+ -+ +-----+ +-----+
+
+ +----- | | +-+ +-----+ +-----+ -+ +-+ +-+ +-----+ +-+---+ +-----+
+
+ | ---+---+ | | | -----+-----+ | | +-----+ +-+--- | +-----+ +
+
+----- | +-+ +-----+ +-----+ | -----+-----+ -+ +-+ +-+ +-----+ | +
+
+ -+---+ +-+ -+ +-----+ -+---+ +-+--- | -+---+ | +-+ -+---+ | +-----+
+
+ | | | | +-+ +-----+ +-+ +-+-----+-----+ +-+ +-----+ +-----+
+
+ -+---+ +-+ | -----+ +-+ | -----+-----+ +-+ | +-+ +-----+ +
+
+++++

```

- (5 Punkte, wichtig) **Rückblick auf das erste Semester:** Schreiben Sie (für sich) eine kurze Zusammenfassung des Stoffs des ersten Semesters (diese Zusammenfassung brauchen Sie nicht in eClaus abzugeben). Geben Sie eine Liste von 10 Punkten an, von denen Sie denken, dass sie besonders wichtig sind. Diese Punkte können sich sowohl auf den Stoff der Informatik im ersten Semester beziehen als auch allgemein sein (zum Beispiel in Hinblick auf die kommenden Semester).

Stellen Sie einige Fragen zusammen, die aus Ihrer Sicht noch nicht geklärt wurden.

Fragen können im Forum <http://www.autip.de/forum/viewforum.php?f=236> diskutiert werden. Weitere Informationen zur Vorlesung / Übung unter <http://www.fmi.uni-stuttgart.de/fk/lehre/ws08-09/autip1/>