

# Mathematikaufgaben

## > Informatik

### > Java-Programm

---

**Aufgabe:** Ein Stack ist ein Stapel- oder Kellerspeicher, der häufig in der Informatik und beim Programmieren Verwendung findet. Der Stapel soll dabei die Buchstaben A, B, C, D als Elemente aufnehmen und nach Zwischenspeicherung gemäß dem Prinzip „last in-first out“ wieder abgeben; d.h. das als Letztes dem Speicher übergebene Element (A, B, C oder D) kann aus diesem auch wieder entfernt werden.

Erstelle in der Programmiersprache Java ein entsprechendes Programm zum Speichern und Abrufen einer bis zu zehn Elementen langen Abfolge von Buchstaben A, B, C, D.

**Lösung:** Das nachstehend aufgelistete Programm „Stack mit Buchstaben A-D“ füllt auf und leert einen Stack mit den Buchstaben A bis D durch Anklicken von Buttons in einem Window (Frame). Das Listing des Programms stack.java lautet:

```
/* stack.java */
import java.awt.*;
import java.awt.event.*;

class cStack {
    int ic = 9;
    int top;
    String arrayStack;
    String text;

    public String initStack() {
        top = -1;
        arrayStack = "";
        text = arrayStack + "/" + top;
        return text;
    }

    public String pushStack(String x) {
        if (top < ic) {
            text = arrayStack + "/" + top;
            top++;
            arrayStack = x + arrayStack;
            text = text + "+" + x + "=" + arrayStack + "/" + top;
        }
    }
}
```

```

        if (top == ic) {
            text = arrayStack + "/" + top;
            text = text + "+" + x + "=" + arrayStack + "/" + top;
        }
        return text;
    }
}

public String popStack() {
    if (top > -1) {
        text = arrayStack + "/" + top;
        top--;
        arrayStack = arrayStack.substring(1);
        text = text + "-" + "=" + arrayStack + "/" + top;
    }
    if (top == -1) {
        text = arrayStack + "/" + top;
        text = text + "-" + "" + "=" + arrayStack + "/" + top;
    }
    return text;
}
}

public class stack extends Frame {
    // Variablen für Komponenten deklarieren

    // Ereignisempfänger
    ButtonListener butLis = new ButtonListener();

    // Bezeichnungsfeld für Anzeige der Rechnung
    Label nlabel = new Label();

    // Grundfläche für Tastatur
    Panel tpanel = new Panel();
    // Grundfläche für Rechnung
    Panel npanel = new Panel();

    // Initialisieren Stack
    cStack obj = new cStack();

    public stack() {
        super();

        // Hauptfenster einrichten
        setTitle("Stack mit Buchstaben A-D");
        setBackground(Color.LIGHT_GRAY);

        // Layout für Container festlegen
        setLayout(new BorderLayout());

        // Tasten für Telefontastatur:
        Button taste1 = new Button("A");
        Button taste2 = new Button("B");
        Button taste3 = new Button("C");
        Button taste4 = new Button("D");
        Button tsubt = new Button("-");
        Button tclear = new Button("X");

        // Tasten in Panel für Tastatur einfügen:
        tpanel.setLayout(new GridLayout(0, 3, 8, 8));
        tpanel.add(taste1);
        tpanel.add(taste2);
        tpanel.add(taste3);
        tpanel.add(taste4);
        tpanel.add(tsubt);
        tpanel.add(tclear);

        // Ereignisempfänger registrieren
        taste1.addActionListener(butLis);
        taste2.addActionListener(butLis);
        taste3.addActionListener(butLis);
        taste4.addActionListener(butLis);
        tsubt.addActionListener(butLis);
        tclear.addActionListener(butLis);
    }
}

```

```

// Bezeichnungsfeld für das Grundrechnen
npanel.setLayout(new GridLayout());

// Anfängliches Bezeichnungsfeld anzeigen
nlabel.setText(obj.initStack());

// Ausrichtung für Bezeichnungsfeld festlegen
nlabel.setAlignment(Label.LEFT);

// Bezeichnungsfeld in Panel einfügen
npanel.add(nlabel);

// Grundflächen in Frame hinzufügen
add(tpanel, BorderLayout.CENTER);
add(npanel, BorderLayout.SOUTH);

// Fenster schließen
addWindowListener(new WindowLauscher());
}

// Ereignisbehandlungsmethoden für Komponenten
// Schaltflächenklicks auswerten
class ButtonListener implements ActionListener {
    public void actionPerformed(ActionEvent e) {
        String s;

        // Befehlszeichenfolge von Taste übernehmen
        s = e.getActionCommand();

        if ((s == "A") || (s == "B") || (s == "C") || (s == "D")) {
            nlabel.setText(obj.pushStack(s));
        }

        if (s == "-") {
            nlabel.setText(obj.popStack());
        }

        // Falls "X" gedrückt, alles löschen
        if (s == "X") {
            nlabel.setText(obj.initStack());
        }
    }
}

// Fenster und Anwendung schließen
protected static final class WindowLauscher extends WindowAdapter {
    public void windowClosing(WindowEvent e) {
        System.exit(0);
    }
}

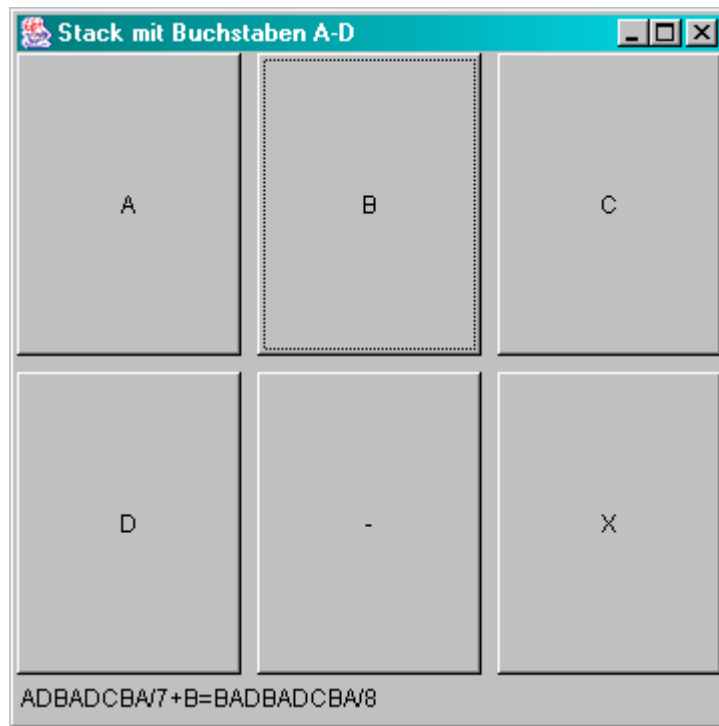
public static void main(String args[]) {
    // Fenster erzeugen und anzeigen
    stack hauptfenster = new stack();
    hauptfenster.setSize(360,360);
    hauptfenster.setLocation(200,300);
    hauptfenster.show();
}
}

```

Für das Funktionieren des Programms über ein Window (Frame) müssen die Bibliotheken `java.awt.*` und `java.awt.event.*` bereit gestellt werden. Das Hauptprogramm `main` erzeugt dann ein Hauptfenster (Window), das als Frame erzeugt wird. Im Frame werden eingabeseitig über ein Panel die Buttons „A“, ... „D“, „-“ und „X“ erzeugt (dreispaltiges `GridLayout` mit Abständen zwischen den Buttons) und mit den entsprechenden Ereignissen (Anklicken eines Buttons) verbunden (Eventhandler). Ein Panel für die Ausgabe von ursprünglichem Stack, Stackoperation und neuem Stack bildet das Umfeld zur Anzeige der Operation in der Form „<Stack alt>/<alte Position des oberen Stackelements><Operation +A-D bzw. ->=<Stack neu>/<neue Position des oberen Stackelements>“ (String). Die an die Ereignisbehandlung angebundene Auswertung der Schaltflächenklicks erzeugt den Ausgabestring in `public`-Routinen. Die Stackgröße ist auf die Länge von 10 begrenzt.

Der im Programm erfolgten Definition des Frame entsprechend kann dieser als Anwendung unter Windows vergrößert oder verkleinert bzw. minimiert oder geschlossen werden. (Quadratisch defi-

nierter) Frame und Programm haben dann das folgende Aussehen:



[www.michael-buhlmann.de](http://www.michael-buhlmann.de) / 11.2012-12.2023 / Aufgabe 1952