



# Universität Karlsruhe (TH)

## Institut für Innovatives Rechnen und Programmstrukturen (IPD)

Real Life Programming (Praktikum im SS 2005) <http://www.info.uni-karlsruhe.de/>  
Dipl.-Inf. Michael Beck [beck@ipd.info.uni-karlsruhe.de](mailto:beck@ipd.info.uni-karlsruhe.de)  
Dipl.-Inform. Rubino Geiß [rubino@ipd.info.uni-karlsruhe.de](mailto:rubino@ipd.info.uni-karlsruhe.de)  
Dipl.-Inform. Sebastian Hack [hack@ipd.info.uni-karlsruhe.de](mailto:hack@ipd.info.uni-karlsruhe.de)

Übungsblatt 4

Ausgabe: 03.05.2005

Besprechung: 24.05.2005

### Aufgabe 1: Knobeln

#### 1.1 dottool (7 Punkte)

Reimplementiere folgendes UNIX-Skript<sup>1</sup> in c-90. Du hast freie Wahl der Implementierungswerkzeuge – insbesondere ist lex und yacc zulässig. Das Ziel ist möglichst effizienten und leicht wartbaren Code zu erstellen.

```
#!/usr/bin/awk -f

function put( c , s ) {
    if (!(c=="") && (s==""))
        printf("%s%s%s; ", templatePre[c], dequote(s), templatePost[c]);
}

function dequote(s) {
    gsub(/\\{/, "{", s); gsub(/\\}/, "}", s); gsub(/\\%/, "%", s);
    gsub(/\\-$/, "-", s); gsub(/\\\\/, "\\ ", s);
    return s;
}

BEGIN { templatePre["$"] = "WriteLn";
        templatePre["%"] = "WriteString(\""; templatePost["%"] = "\")";
}

/^\\.\\.\/ {
    ctrl = substr($0,3,1);
    str = substr($0,4);
    if (match(str, /^(^[^\\])((\\))*/)) {
        templatePre[ctrl] = dequote(substr(str,1,RSTART+RLENGTH-2));
        templatePost[ctrl] = dequote(substr(str,RSTART+RLENGTH));
    } else {
        templatePre[ctrl] = dequote(str);
    }
    printf("\n"); next;
}

/^\.\/ { str = substr($0,2);
    while ( match(str, /((\\))|(\{)|([^\{]*)*/)) {
        put("%", substr(str,1,RSTART+RLENGTH-2));
        ctrl = substr(str,RSTART+RLENGTH,1);
        str = substr(str,RSTART+RLENGTH+1);
        match(str, /((\\))|(\})|([^\}])*/));
        put(ctrl, substr(str,1,RLENGTH-1));
        str = substr(str,RSTART+RLENGTH);
    }

    if ( !match(str, /^(^[^\\])((\\))*/)) {
        put("%", str); put("$", "");
    } else put("%", substr(str,1,RSTART+RLENGTH-2));
    printf("\n"); next;
}

{ print; }
```

#### 1.2 Compilerbug (3 Punkte)

Ein Übersetzer erzeugt aus dem Programm <http://www.info.uni-karlsruhe.de/lehre/2005SS/rlp/material/xxx.c> falschen Assembler-Code <http://www.info.uni-karlsruhe.de/lehre/2005SS/rlp/material/xxx.s>. Was geht schief? Was ist zu ändern, damit dieser Assembler-Code doch noch funktioniert?

<sup>1</sup><http://www.info.uni-karlsruhe.de/lehre/2005SS/rlp/material/dottool>