

Aufgabenblatt 4

Aufgabe 1

In ein Char Array mit einer festgelegten Größe von 1024 Zeichen wird über die Funktion `cin.getline` die Eingabe des Nutzers eingelesen.

```
char *MiniShell::readLine() {  
    int bufsize = 1024;  
    char *line = new char[bufsize];  
    cin.getline(line, bufsize);  
  
    return line;  
}
```

Die Eingabezeile wird mit dem Befehl `strtok` in einzelne Argumente zerlegt.

```
char *token = strtok(line, sep);
```

Umgebungsvariablen werden durch die Methode `getenv` gesucht und anschließend ersetzt.

```
char *env = getenv(args[position] + 1);  
if (env != nullptr) {  
    args[position] = env;  
}
```

Aufgabe 2 / 3 / 4

```
void MiniShell::launch(char **args) {  
    // Neuer Kindprozess per fork() erzeugen  
    pid_t pid = fork();  
  
    if (pid == -1) {  
        cerr << "Fork fehlgeschlagen!" << endl;  
        return;  
    } else if (pid == 0) {  
        if (execvp(args[0], args) < 0) {  
            cerr << "Befehl konnte nicht ausgeführt werden!" << endl;  
        }  
        exit(0);  
    } else {  
        // Elternprozess wartet auf Beendigung des Kindprozesses  
        wait(nullptr);  
        return;  
    }  
}
```

Es wird ein Kindprozess mittels `fork` erstellt. Danach wird mit der Funktion `execvp` mit den einzelnen unterteilten Argumenten ein Befehl ausgeführt. Während der Ausführung wartet der Elternprozess auf die Beendigung des Kindprozesses.

Tests

Test 1: Es wurde eine Umgebungsvariable *t* erstellt (*export*), welche den Namen eines Verzeichnisses trägt. Anschließend wird geprüft, ob diese erfolgreich gesetzt wurde, indem diese mit *showenv* ausgegeben wird. Zuletzt wird die Variable in einem Build-In Befehl *cd* verwendet:

```
<Pfad>$ export t=Testing
<Pfad>$ showenv t
Testing
<Pfad>$ cd $t
<Pfad>/Testing$
```

Test 2: Es wurde der *touch*-Befehl ausgeführt mit mehreren Argumenten:

```
<Pfad>$ touch t1.txt t2.txt
<Pfad>$ ls
t1.txt t2.txt
```

Test 3: Es wurde der Befehl *ls -a/s* ausgeführt und absichtlich mehrere Leerzeichen zwischen dem Befehl und dem Argument hinzugefügt:

```
<Pfad>$ ls          -a/s
total 552...
```