

Short Report

I started and completed this project with the help of previous prelabs and labs. It's quite similar to them.

This project's main goal is to convert shell script language to perl script language.

My project is not perfect but it is capable of making the main goal possible for most of the cases.

It can convert all of the examples(not erroneous) that are given without any errors. (Except Example 5. It converts some if statement wrong.)

Some convertible examples are given below:

- Shell script predictive:

```
#!/bin/sh
```

- Any comments:

```
# Hello World Program
```

- Expressions like:

```
a=5
b=$((3+4))
c=$((a+($b/5)))
d=$((c+10))
e=$((($d+10)*$e))
f=$((($f*$g)-($h+$i))
g=$((($j*$k)*($l-100/($m-$n)))
.
..
...
```

- Echo statements like:

```
echo $a
echo "a = $b and z= $z"
echo 'Prints a*b'
echo $((a*$b))
.
..
...
```

- If – Else statements like:

```
if [ $a -(here can be <,>,<=,>=,== etc.) 200 ]
then
    some statements...
    if [ $b -(here can be <,>,<=,>=,== etc.) 10 ]
    then
        some statements..
    fi
    elif [ $b -(here can be <,>,<=,>=,== etc.) 200]
    then
        some statements..
elif [ $c -(here can be <,>,<=,>=,== etc.) 200]
then
    some statements...
else
    some statements...
fi
```

What it can't do?

It can't detect some syntax errors if there are any. (Example: Paranthesis check)

However it can detect if – else statements syntax errors. Like if there is an missing “then”, or “fi”, it will tell you the line that is not correct.

It can't detect the blank space after the condition and before the “]”.

It can't convert if-else statements if it is nested in more than 3 if statements. (It wouldn't convert correctly.)

My Progression

I started the project about 1-1.5 weeks ago. At first, i completed the lex file. I had some errors while compiling it. The error was about yylval values and it couldn't convert to string or int correctly. That took me 1-2 days to solve the problem.

Yacc file is the real issue. At first i used %token for everything we created in yacc. Of course that was not working and make me stop after a lot of hours of searching to find the issue. When i realized that i need to use %type for the ones we created in Yacc, i was finally be able to move on to the line error check.

I did some recursive statements and some are one by one. I spend way too much time to make everything recursive so that any code can be converted correctly. I everytime i tried something i had the print order issue. For example, 3+4 prints first before print statement. And something like that shows up: “3+4;printf”. I decided to make some things not recursive and move on to the next examples. If-Else statements was harder than i thought. The curly paranthesis, it's tab space, nested statements took me a while to finish the project.

I believe i have learned a lot in this project. It was hard and entertaining at the same time.