

Laboratorio #1

María Marta Ramirez Gil 21342

Universidad del Valle de Guatemala

Construcción de Compiladores

Julio 2024

Repositorio:

<https://github.com/mariaRam2003/Compiladores2/tree/Laboratorio1>

Ejecucion de Docker

1.

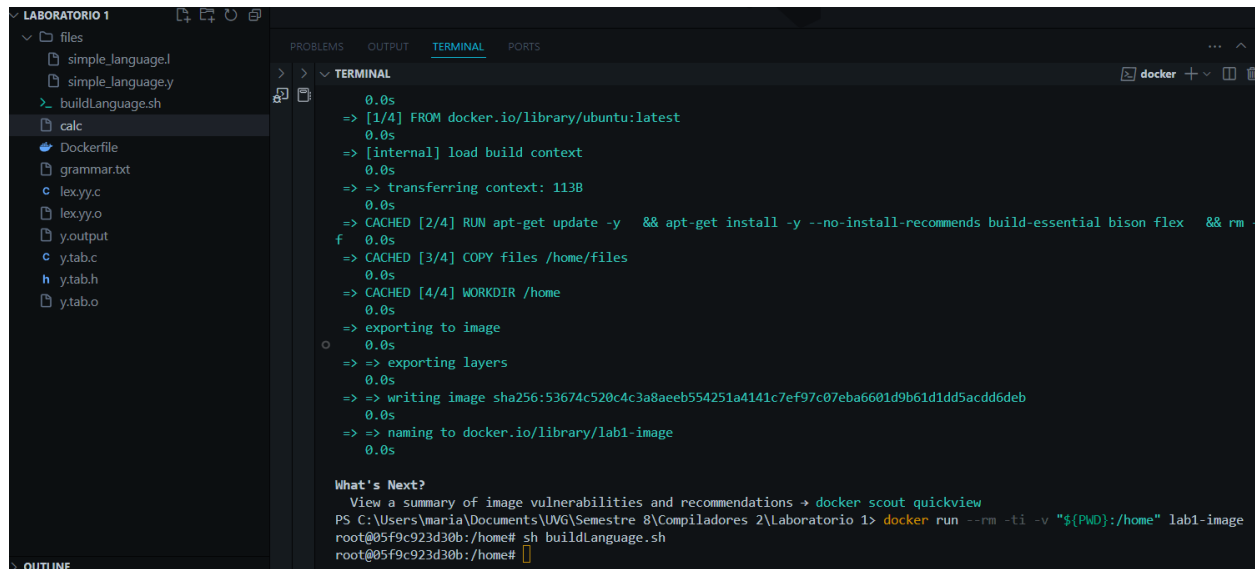
```
PS C:\Users\maria\Documents\UVG\Semestre 8\Compiladores 2\Laboratorio 1> docker build --rm . -t lab1-image
[+] Building 0.1s (9/9) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile             0.0s
=> => transferring dockerfile: 244B                             0.0s
=> [internal] load metadata for docker.io/library/ubuntu:latest 0.0s
=> [internal] load .dockerignore                               0.0s
=> => transferring context: 2B                                   0.0s
=> [1/4] FROM docker.io/library/ubuntu:latest                  0.0s
=> [internal] load build context                                0.0s
=> => transferring context: 113B                                 0.0s
=> CACHED [2/4] RUN apt-get update -y && apt-get install -y --no-install-recommends build-essential bison flex && rm -rf 0.0s
=> CACHED [3/4] COPY files /home/files                         0.0s
=> CACHED [4/4] WORKDIR /home                                 0.0s
=> exporting to image                                          0.0s
=> => exporting layers                                          0.0s
=> => writing image sha256:53674c520c4c3a8aeb554251a4141c7ef97c07eba6601d9b61d1dd5acdd6deb 0.0s
=> => naming to docker.io/library/lab1-image                   0.0s

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview
```

2.

```
PS C:\Users\maria\Documents\UVG\Semestre 8\Compiladores 2\Laboratorio 1> docker run --rm -ti -v "${PWD}:/home" lab1-image
root@05f9c923d30b:/home#
```

3.



Actividades

1. Asignacion de Variables

```
root@05f9c923d30b:/home# ./calc
a = 5
b = 2
Assign a = 5
a + b:
Assign b = 2
7
```

Nota: Asigna las variables correctamente y realiza la operación aritmética simple.

2. Operación aritmética simple

```
root@05f9c923d30b:/home# ./calc
1256 + 15:
1271
```

3. Experimente con expresiones más complejas y verifique que el compilador las procese correctamente.

```
root@05f9c923d30b:/home# ./calc
2 * (3 + 4):
10
5 + 12 * (15 * 5):
905
78 / (23 + 108) + 11 * (1 + 8):
130
```

Nota: En este caso el compilador no ha resuelto correctamente 2 de estos problemas.

4. Modifique el lenguaje para incluir la asignación de variables con expresiones aritméticas.

```
root@05f9c923d30b:/home# ./calc
b = 4 + 2:
Assign b = 6
```

5. Agregue manejo de errores al compilador para detectar tokens inválidos en el programa fuente

```
root@05f9c923d30b:/home# ./calc
#$+(&%:
syntax error
Error: Token inválido
Error: Token inválido
Error: Token inválido
!!@@:
Error: Token inválido
! - 8:
Error: Token inválido
```

6. Experimente con la precedencia de operadores en el lenguaje y observe como afecta la generación del árbol sintáctico.

6.1

```
%right '='  
%left '+' '-'  
%left '*' '/'
```

```
root@05f9c923d30b:/home# ./calc  
a = 5 + 3 * 2:  
NUMBER: 5  
NUMBER: 3  
NUMBER: 2  
MULTIPLY: (3 * 2)  
SUM: (5 + 6)  
Assign a = 11
```

6.2

```
%right '+' '-' '/'  
%left '='  
%left '*'
```

```
root@05f9c923d30b:/home# ./calc  
a = 5 + 3 * 2:  
NUMBER: 5  
NUMBER: 3  
NUMBER: 2  
MULTIPLY: (3 * 2)  
SUM: (5 + 6)  
Assign a = 11
```

6.3

```
%right '='  
%left '*' '/'  
%left '+' '-'
```

```
root@05f9c923d30b:/home# ./calc  
a = 5 + 3 * 2:  
Suma: 5 + 3  
Multiplicación: 8 * 2  
Assign a = 16
```

Nota: Con estas impresiones, se puede dar cuenta que para este operación aritmética no hay muchos cambios si se cambia la configuración de precedencia. Sin embargo al realizar un cambio (6.3), se puede observar cómo la configuración de precedencia en Yacc influye directamente en el orden de

evaluación de las operaciones dentro de las expresiones, afectando así la estructura y el resultado del árbol sintáctico generado.