

Code Generation

SHRI VISHNU ENGINEERING
COLLEGE FOR WOMEN

BHIMAVARAM

April 9, 2022

BATCH - 5

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

1. 20B01A05H8-V.RAKSHITHA-CSE
2. 20B01A05H9-V.RESHMA-CSE
3. 20B01A1215-B.BINDU SRI-IT
4. 20B01A1216-B.HARITHA-IT
5. 20B01A04C0-P.KEERTHI-ECE

INTRODUCTION

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

Code Generation is building a translator to convert the postfix arithmetic expression into assembly language code

DESCRIPTION

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

Code Generator is a backend translator for an SIC machine. Input to the translator will be arithmetic expressions in postfix form and the output will be assembly language code.

FLOW DIAGRAM

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

DAY-1

Understood the project description and created Gitlab repository

DAY-2

Work done on retrieving the data from input file

DAY-3

Developed code to generate assembly language from postfix expression

DAY-4

Documentation is done, code is modified and then executed

DAY-5

Presentation by team members and project Submission

MEMBERS CONTRIBUTION

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

- ▶ Bindu Sri - Installed related software and worked on retrieving the input from input.txt file which consists of postfix expression.
- ▶ Keerthi - worked on modifying the inputdata to pass it to the function which generates assembly language.
- ▶ Reshma - Worked on generating assembly language code as output.
- ▶ Rakshitha - Worked on generating assembly language code as output.
- ▶ Haritha - Worked on appending the obtained output in the output.txt file.

CHALLENGES

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

- ▶ for retrieving the input data to the main file
- ▶ understanding and implementing the assembly language

HOW CHALLENGES OVERCAME

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

- ▶ created a seperate input.text file to retrieve it into the main file
- ▶ gaining the knowledge about assembly language through internet

LEARNINGS

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

- ▶ Team collaboration.
- ▶ Preparing presentations in LaTeX.
- ▶ Building projects with Python language.
- ▶ Push files into repository(Gitlab).

TECH STACKS

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

1. Python Programming
2. Pycharm
3. GitLab
4. GitBash

CODE STATS

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

- ▶ Conditional statements and loops (while, for ,if-elif).
- ▶ Functions-
write(),isdigit(),close(),open(),strip(),split()

CODE STATS

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

► Our code consists of 55 lines.

```
1 def printLineCode(i):
2     var = open('output.txt', 'a')
3     temps = 1
4     line = i
5     x = 0
6     if len(line) == 1:
7         if line[0].isdigit():
8             var.write('L #' + line[0] + '\n\n')
9         else:
10            var.write('L ' + line[0] + '\n\n')
11    else:
12        while len(line) != 1:
13            if line[x+2] in "+-*/":
14                op = line[x+2]
15                num1 = '' + line[x]
16                num2 = '' + line[x+1]
17                placeholder = ''
18
19                if num2.isdigit():
20                    placeholder = '#'
21
22                if num1.isdigit():
23                    var.write('L #' + num1 + '\n\n')
24                else:
25                    var.write('L ' + num1 + '\n\n')
26
27                if op == '+':
28                    var.write('A ' + placeholder + num2 + '\n\n')
29
30                    elif op == '-':
31                        var.write('S ' + placeholder + num2 + '\n\n')
32
33                    elif op == '*':
34                        var.write('M ' + placeholder + num2 + '\n\n')
35
36                    elif op == '/':
37                        var.write('D ' + placeholder + num2 + '\n\n')
38                    else:
39                        var.write('N ' + placeholder + num2 + '\n\n')
40
41                var.write('ST $' + str(temps) + '\n\n')
42                line[x] = '$' + str(temps)
43                temps += 1
44                line.pop(x + 1)
45                line.pop(x + 1)
46                x = 0
47            else:
48                x += 1
49        var.close()
50
51    open('output.txt', 'w').close()
52    file = open('input.txt', 'r')
53    for line in file:
54        line = line.strip()
55        line = line[0:len(line)]
56        tokens = line.split(" ")
57        printLineCode(tokens[0:len(line)])
58    file.close()
```

output

output.txt ×	
1	L A
2	
3	A B
4	
5	ST \$1
6	
7	L C
8	
9	A D
10	
11	ST \$2
12	
13	L E
14	
15	A F

REPO

Code
Generation

SHRI VISHNU
ENGINEER-
ING
COLLEGE
FOR WOMEN

► GitLab link

[https://gitlab.com/vrakshitha2468/
codegeneration-wise-batch-5.git](https://gitlab.com/vrakshitha2468/codegeneration-wise-batch-5.git)

