**Compiler Design Lab**

**Practical-5**

**Name: Yash Patni**

**Roll: 71**

**Batch: A4**

**Aim: Write a program to generate three address code for the given language construct using SDTS.**

**Code:**

from prettytable import PrettyTable

def generate\_three\_address\_code(code\_lines):

    final\_code = []

    temp\_counter = 1

    do\_index = None

    instruction\_pointer = 100

    for line in code\_lines:

        line = line.strip()

        if 'do' in line:

            do\_index = instruction\_pointer

            instruction\_pointer += 1

            continue

        elif 'while' in line:

            conditions = line[line.find('(') + 1:line.rfind(')')].split('||')

            for condition in conditions:

                final\_code.append(f'if {condition.strip()} goto {do\_index}')

                final\_code.append('goto')

                instruction\_pointer += 1

            final\_code.append("End")

            instruction\_pointer += 1

        else:

            statements = line.split(';')

            for statement in statements:

                if '=' in statement:

                    lhs, rhs = statement.split('=')

                    lhs = lhs.strip()

                    rhs = rhs.strip()

                    if '\*' in rhs or '/' in rhs:

                        temp\_var = f't{temp\_counter}'

                        final\_code.append(f'{temp\_var} = {rhs}')

                        final\_code.append(f'{lhs} = {temp\_var}')

                        temp\_counter += 1

                    elif '+' in rhs or '-' in rhs:

                        temp\_var1 = f't{temp\_counter}'

                        temp\_counter += 1

                        temp\_var2 = f't{temp\_counter}'

                        final\_code.append(f'{temp\_var1} = {rhs}')

                        final\_code.append(f'{temp\_var2} = {temp\_var1}')

                        final\_code.append(f'{lhs} = {temp\_var2}')

                        temp\_counter += 1

                    elif rhs.isdigit():

                        final\_code.append(f'{lhs} = {rhs}')

                    else:

                        temp\_var = f't{temp\_counter}'

                        final\_code.append(f'{temp\_var} = {rhs}')

                        final\_code.append(f'{lhs} = {temp\_var}')

                        temp\_counter += 1

                    instruction\_pointer += 1

                elif statement.strip():

                    final\_code.append(statement.strip())

                    instruction\_pointer += 1

    return final\_code

def read\_code\_from\_file(file\_name):

    with open(file\_name) as f:

        code\_lines = f.readlines()

    return [line.strip() for line in code\_lines if line.strip()]

def display\_three\_address\_code(three\_address\_code):

    print('\nThe Three Address Code Generated is:')

    table = PrettyTable()

    table.align = "l"

    table.field\_names = ['Index','Code']

    for i, code\_line in enumerate(three\_address\_code):

        table.add\_row([100 + i, code\_line])

    print(table)

code\_lines = read\_code\_from\_file('/content/code.txt')

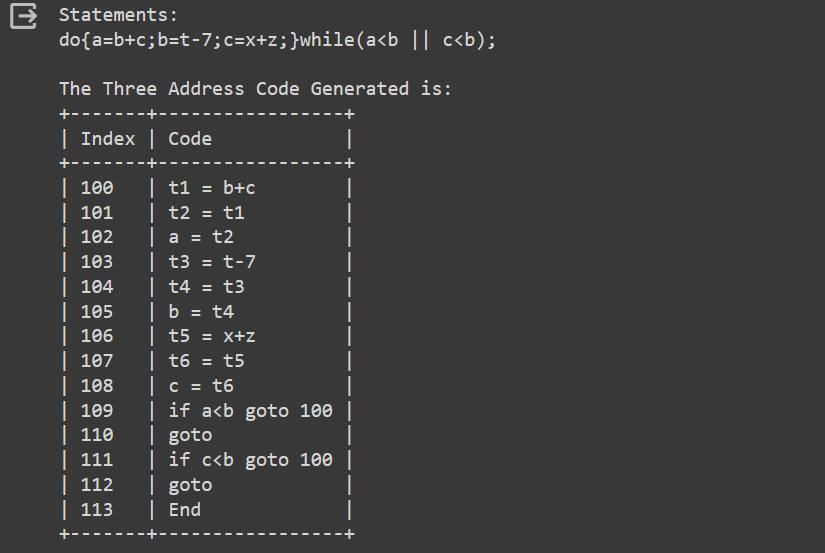
print('Statements:')

print(''.join(code\_lines))

three\_address\_code = generate\_three\_address\_code(code\_lines)

display\_three\_address\_code(three\_address\_code)

**Output:**

****

**Conclusion:** In this practical we successfully generated 3 address code for do-while loop using sdts.