

Assignment No. - 11

Class – TYBTech - C

Roll No. – 51

Name of Student – Kaustubh Upare

Aim – Write a program to implement code optimization for a given block of three address codes.

Program –

```
def is_temp(var):
    return var.startswith("t")

def parse_instruction(instr):
    parts = instr.replace("=", " = ").replace("+", " + ").replace("-", " - ").replace("*", "
* ").replace("/", " / ").split()
    if len(parts) == 5:
        return (parts[0], parts[2], parts[3], parts[4])
    elif len(parts) == 3:
        return (parts[0], parts[2], None, None)
    else:
        return None

def optimize_TAC(tac):
    optimized = []
    expr_table = {}
    used_vars = set()

    for instr in tac:
        if '=' in instr:
            left, right = instr.split('=')
            left = left.strip()
            right = right.strip()
            for token in right.split():
                if token.isidentifier() and not is_temp(token):
                    used_vars.add(token)

    for instr in tac:
        parsed = parse_instruction(instr)
        if not parsed:
```

```
optimized.append(instr)
continue
```

```
result, arg1, op, arg2 = parsed
```

```
if op:
    key = (arg1, op, arg2)
    if key in expr_table:
        optimized.append(f'{result} = {expr_table[key]}')
    else:
        expr_table[key] = result
        optimized.append(instr)
    else:
        optimized.append(instr)
```

```
final_code = []
for instr in optimized:
    left = instr.split('=')[0].strip()
    if not is_temp(left) or left in used_vars:
        final_code.append(instr)
```

```
return final_code
```

```
tac = []
n = int(input("Enter the number of TAC instructions:
"))
print("Enter the TAC instructions (e.g., t1 = a +
b):")
for _ in range(n):
    tac.append(input())
```

```
print("\nOriginal TAC:")
for line in tac:
    print(line)
```

```
optimized = optimize_TAC(tac)
```

```
print("\nOptimized TAC:")
for line in optimized:
    print(line)
```

Input –

```
Enter the number of TAC instructions: 6
Enter the TAC instructions (e.g., t1 = a + b):
t1 = a + b
t2 = a + b
t3 = t1 + c
t4 = t2 + c
t5 = t4 + d
x = t5
```

Output –

```
Enter the number of TAC instructions: 6
Enter the TAC instructions (e.g., t1 = a + b):
t1 = a + b
t2 = a + b
t3 = t1 + c
t4 = t2 + c
t5 = t4 + d
x = t5
```

Original TAC:

```
t1 = a + b
t2 = a + b
```

```
t3 = t1 + c
t4 = t2 + c
t5 = t4 + d
x = t5
```

Optimized TAC:

```
t1 = a + b
t2 = t1
t3 = t1 + c
t4 = t2 + c
t5 = t4 + d
x = t5
```