Department of Computer Science and Engineering

S.G.Shivanirudh , 185001146, Semester VI

23 April 2021

UCS1602 - Compiler Design

Exercise 8: Code optimisation using C

Objective:

Develop a C program to optimise the code generated as intermediate code.

Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void optimize(char *s) {

    // addition
    if(s[3]=='+'){
        if(s[2]=='0'||s[4]=='0'){
            if(s[0]==s[2]||s[0]==s[4]){
```

```
11                  printf("\n");
12              }
13              else{
14                  printf("%c%c%c\n",s[0],s[1],s[2]=='0'?s[4]:s
    [2]);
15              }
16          }
17          else{
18              printf("%s",s);
19          }
20      }
21      else if(s[3]=='*'){
22          if(s[2]=='1'||s[4]=='1'){
23              if(s[0]==s[2]||s[0]==s[4]){
24                  printf("\n");
25              }else{
26                  printf("%c%c%c\n",s[0],s[1],s[2]=='1'?s[4]:s
    [2]);
27              }
28          }
29          if(s[2]==s[4]){
30              printf("%c%c%c+%c\n",s[0],s[1],s[2],s[4]);
31          }
32      }
33      else if(s[3]=='-'){
34          if(s[2]=='0'||s[4]=='0'){
35              if(s[0]==s[2]||s[0]==s[4]){
36                  printf("\n");
37              }else{
38                  printf("%c%c%c%c\n",s[0],s[1],s[2]=='0'?'-':'
    ',s[2]=='0'?s[4]:s[2]);
39              }
40          }
41          if(s[2]==s[4]){
42              printf("%c%c%c+%c\n",s[0],s[1],s[2],s[4]);
43          }
44      }
45      else if(s[3]=='/'){
46          if(s[4]=='1'){
47              if(s[0]==s[2]){
48                  printf("\n");
49              }else{
50                  printf("%c%c%c\n",s[0],s[1],s[2]);
51              }
52          }
```

```
53          if(s[2]=='0'){
54              printf("%c%c%c\n",s[0],s[1],'0');
55          }
56      }
57      else if(s[2]=='p'){
58          if(s[8]=='2'){
59              printf("%c%c%c*%c\n",s[0],s[1],s[6],s[6]);
60          }else{
61              printf("%s",s);
62          }
63      }
64  }
65
66  int main(int argc, char *argv[]){
67      FILE *fp;
68      fp = fopen(argv[1], "r");
69      int i = 0;
70      int tot = 0;
71      char lines[100][100];
72      while(fgets(lines[i], 100, fp)) {
73          lines[i][strlen(lines[i])] = '\0';
74          i++;
75      }
76      tot = i;
77
78      for(i = 0; i < tot; ++i) {
79          optimize(lines[i]);
80      }
81  }
```

Input file:

```
x=x+0
y=y*1
x=0+x
x=y+1
y=1*y
x=z+0
x=w*w
x=pow(i,2)
x=pow(i,3)
x=0-y
x=y-0
x=x/1
x=y/1
x=0/x
```

Output:

```
➜ ./a ip.txt



x=y+1

x=z
x=w+w
x=i*i
x=pow(i,3)
x=-y
x= y

x=y
x=0
```

Learning Outcomes:

- Understood the basic idea of Code optimisation.
- Learnt what sort of expressions needed to be simplified.
