A8-Implementation of code optimization techniques

Sneha Sriram Kannan 185001157

23-04-2021

1 Code

Optv1.c - Algebraic Expression Optimization (+,-,*,/)

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

void CodeOptimizer();

int main()
{
    CodeOptimizer();
    return 0;
}
void CodeOptimizer()
{
    char file[10][128];
    int i = 0;
    FILE *fd = fopen("Code.txt", "r");
    while (fgets(file[i], sizeof(file[i]), fd))
        i++;
    for (int j = 0; j < i; j++)
    {
        if (file[j][3] == '+') //addition
        {
            if (file[j][2] == '0')
            {
                if (file[j][0] != file[j][4])
                    printf("%c=%c\n", file[j][0], file[j][4]);
            }
            else if (file[j][4] == '0')
            {
                if (file[j][0] != file[j][2])
```

```
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == '*') //multiplication
        {
            if (file[j][2] == '1')
            {
                if (file[j][0] != file[j][4])
                    printf("%c=%c\n", file[j][0], file[j][4]);
            }
            else if (file[j][4] == '1')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == '-') //subtraction
        {
            if (file[j][4] == '0')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else if (file[j][2] == '0')
            {
                printf("%c=-%c\n", file[j][0], file[j][4]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == '/') //division
        {
            if (file[j][4] == '1')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else
                printf("%s", file[j]);
        }
        else
        {
            printf("%s", file[j]);
        }
    }
```

```
}
```

Optv2.c - Algebraic Expression and Strength Reduction

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

void CodeOptimizer();

int main()
{
    CodeOptimizer();
    return 0;
}
void CodeOptimizer()
{
    char file[10][128];
    int i = 0;
    FILE *fd = fopen("Code2.txt", "r");
    while (fgets(file[i], sizeof(file[i]), fd))
        i++;
    for (int j = 0; j < i; j++)
    {
        if (file[j][3] == '+') //addition
        {
            if (file[j][2] == '0')
            {
                if (file[j][0] != file[j][4])
                    printf("%c=%c\n", file[j][0], file[j][4]);
            }
            else if (file[j][4] == '0')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == '*' && file[j][4] != '*') //
   multiplication
        {
            if (file[j][2] == '1')
            {
                if (file[j][0] != file[j][4])
                    printf("%c=%c\n", file[j][0], file[j][4]);
```

```
            }
            else if (file[j][4] == '1')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == '-') //subtraction
        {
            if (file[j][4] == '0')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else if (file[j][2] == '0')
            {
                printf("%c=-%c\n", file[j][0], file[j][4]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == '/') //division
        {
            if (file[j][4] == '1')
            {
                if (file[j][0] != file[j][2])
                    printf("%c=%c\n", file[j][0], file[j][2]);
            }
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == file[j][4] && file[j][4] == '*')
        {
            if (file[j][5] == '2')
                printf("%c=%c*%c\n", file[j][0], file[j][2],
file[j][2]);
            else
                printf("%s", file[j]);
        }
        else if (file[j][3] == file[j][4] && file[j][4] == '*')
        {
            if (file[j][5] == '2')
                printf("%c=%c*%c\n", file[j][0], file[j][2],
file[j][2]);
            else
                printf("%s", file[j]);
        }
```

```
89          else if (file[j][2] == 'p' && file[j][3] == 'o' && file
    [j][4] == 'w' && file[j][8] == '2')
90          {
91              printf("%c=%c*%c\n", file[j][0], file[j][6], file[j
    ][6]);
92          }
93          else
94          {
95              printf("%s", file[j]);
96          }
97      }
98  }
```

2 Output Screenshots

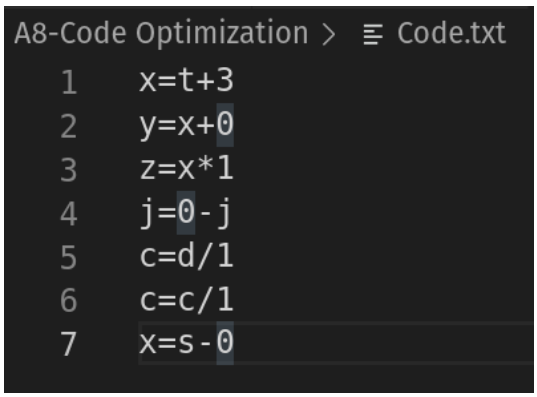

Figure 1: Input Version 1

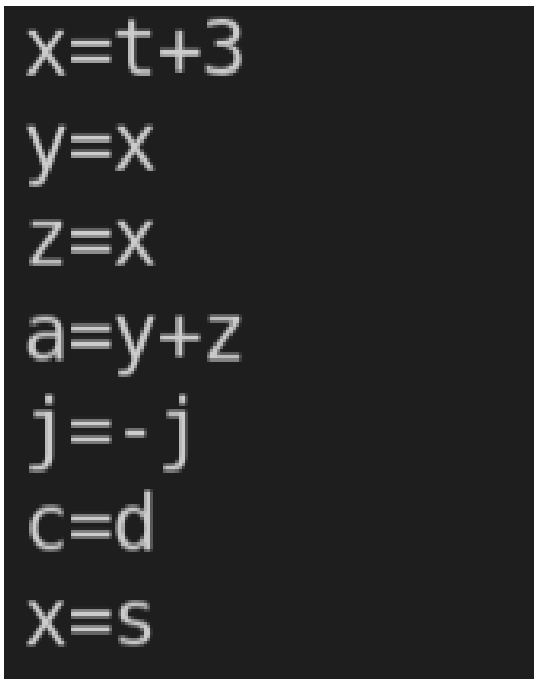

Figure 2: Output Version 1

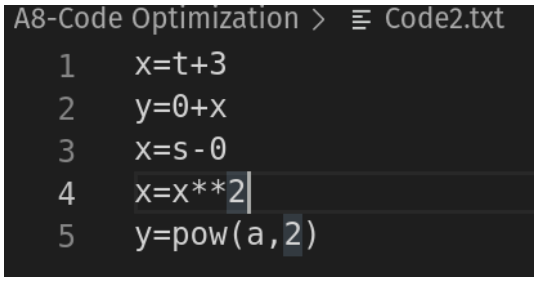

Figure 3: Input Version 2

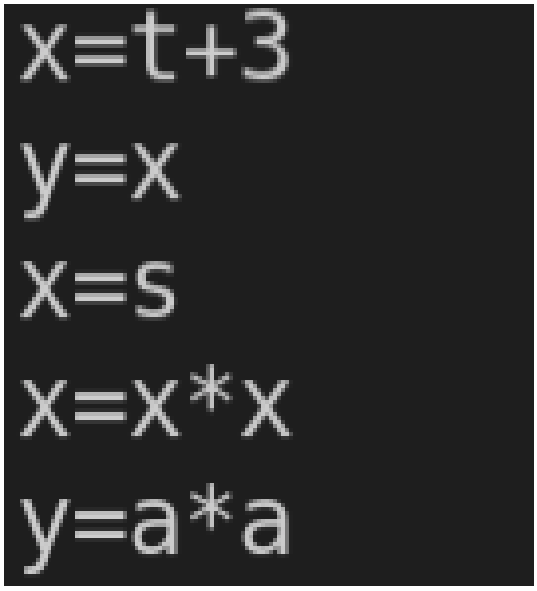

Figure 4: Output Version 2