2CS701- Compiler Construction

Practical 9

Aim: To implement Assembly code generator

Author: Darshil Maru 20BCE514

Guide: Prof. Deepti Saraswat

Code:

```
#include <stdio.h>
#include <stdio.h>
#include <string.h>
void main() {

   char icode[10][30], str[20], opr[10];
   int i = 0;

   printf("\n Enter the set of intermediate code (terminated by exit):\n");

   do
   {
      scanf("%s", icode[i]);
   } while (strcmp(icode[i++], "exit") != 0);

   printf("\n target code generation");
   printf("\n**********");

   i = 0;
   do {
      strcpy(str, icode[i]);
      switch (str[3]) {
         case '+':
            strcpy(opr, "ADD ");
            break;
         case '-':
            strcpy(opr, "SUB ");
            break;
         case '*':
            strcpy(opr, "MUL ");
            break;
         case '/':
            strcpy(opr, "DIV ");
            break;
```

```
        }
        printf("\n\tMov %c,R%d", str[2], i);
        printf("\n\t%s%c,R%d", opr, str[4], i);
        printf("\n\tMov R%d,%c", i, str[0]);

    } while (strcmp(icode[++i], "exit") != 0);

    return;
}
```

Output:

```
/tmp/urbTR355Qy.o
Enter the set of intermediate code (terminated by exit):
a=b*c
b=c+d
c=a-c
exit
target code generation
**********
    Mov b,R0
    MUL c,R0
    Mov R0,a
    Mov c,R1
    ADD d,R1
    Mov R1,b
    Mov a,R2
    SUB c,R2
    Mov R2,c
```