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

int i = 1, j = 0, no = 0, tmpch = 90;
char str[100], left[15], right[15];
void findopr();
void explore();
void fleft(int);
void fright(int);
struct exp {
    int pos;
    char op;
} k[15];

void main(){
    printf("\t\tINTERMEDIATE CODE GENERATION\n\n");
    printf("Enter the Expression :");
    scanf("%s", str);
    printf("The intermediate code:\n");
    findopr();
    explore();
}

void findopr(){
    for (i = 0; str[i] != '\0'; i++)
        if (str[i] == ':') {
            k[j].pos = i;
            k[j++].op = ':';
        }
    for (i = 0; str[i] != '\0'; i++)
        if (str[i] == '/') {
            k[j].pos = i;
            k[j++].op = '/';
        }
    for (i = 0; str[i] != '\0'; i++)
        if (str[i] == '*') {
            k[j].pos = i;
            k[j++].op = '*';
        }
    for (i = 0; str[i] != '\0'; i++)
        if (str[i] == '+') {
            k[j].pos = i;
            k[j++].op = '+';
        }
    for (i = 0; str[i] != '\0'; i++)
        if (str[i] == '-'){
            k[j].pos = i;
            k[j++].op = '-';
        }
}

void explore(){
    i = 1;
    while (k[i].op != '\0') {
        fleft(k[i].pos);
        fright(k[i].pos);
        str[k[i].pos] = tmpch--;
        printf("\t%c := %s%c%s\t\t", str[k[i].pos], left, k[i].op, right);
        printf("\n");
        i++;
    }
    fright(-1);
    if (no == 0) {
        fleft(strlen(str));
        printf("\t%s := %s", right, left);
        exit(0);
    }
    printf("\t%s := %c", right, str[k[--i].pos]);
}

void fleft(int x) {
    int w = 0, flag = 0;
    x--;
    while (x != -1 && str[x] != '+' && str[x] != '*' && str[x] != '=' && str[x] !=
'\0' && str[x] != '-' && str[x] != '/' && str[x] != ':'){
        if (str[x] != '$' && flag == 0){
            left[w++] = str[x];
            left[w] = '\0';
            str[x] = '$';
            flag = 1;
        }
        x--;
    }
}

void fright(int x) {
    int w = 0, flag = 0;
    x++;
    while (x != -1 && str[x] != '+' && str[x] != '*' && str[x] != '\0' && str[x] !=
'=' && str[x] != ':' && str[x] != '-' && str[x] != '/') {
        if (str[x] != '$' && flag == 0){
            right[w++] = str[x];
            right[w] = '\0';
            str[x] = '$';
            flag = 1;
        }
        x++;
    }
}
```