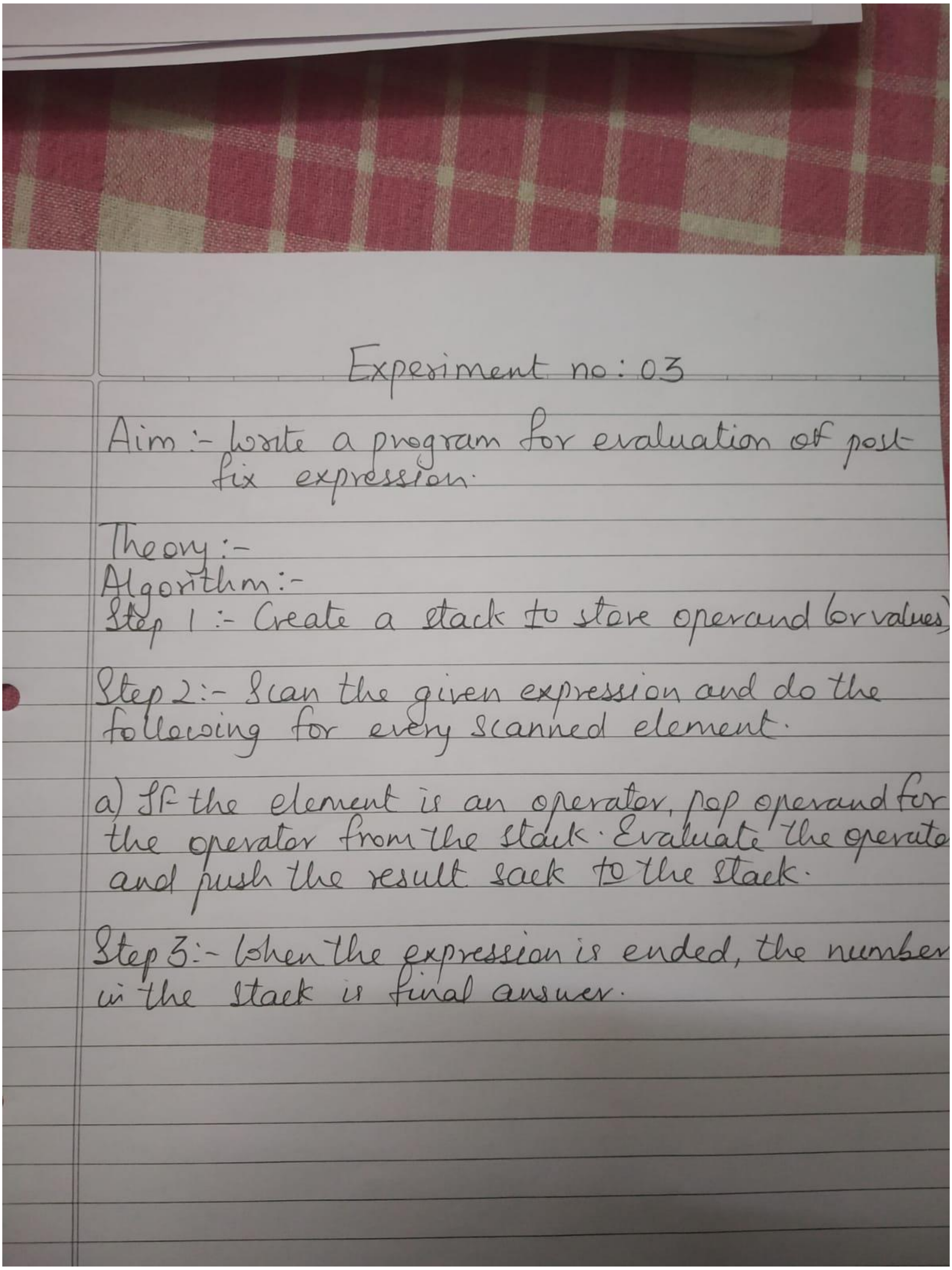
Vivekanand Education Society's
Institute of Technology
Approved by AICTE & Affiliated to University of Mumbai

COMPUTER ENGINEERING

DS ODD SEM 2021-22/EXPERIMENT 3

NAME:- GAURAV AMARNANI (D7A, 67)



Experiment no: 03

Aim :- Write a program for evaluation of post fix expression.

Theory :-

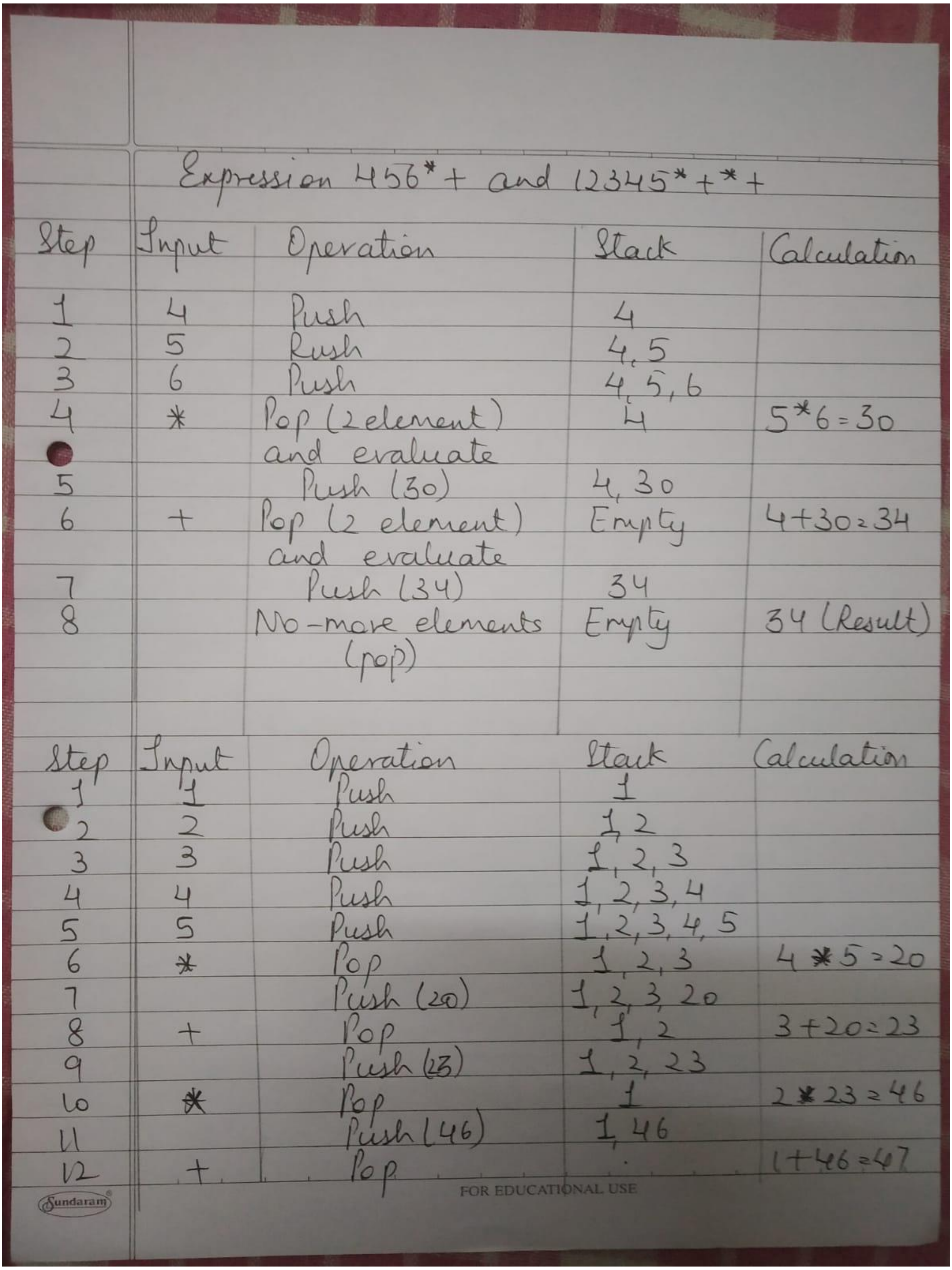
Algorithm :-

Step 1 :- Create a stack to store operand (or values)

Step 2 :- Scan the given expression and do the following for every scanned element.

a) If the element is an operator, pop operand for the operator from the stack. Evaluate the operato and push the result back to the stack.

Step 3 :- When the expression is ended, the number in the stack is final answer.



Expression 456* + and 12345* + * +

Step	Input	Operation	Stack	Calculation
1	4	Push	4	
2	5	Rush	4, 5	
3	6	Push	4, 5, 6	
4	*	Pop (2 element) and evaluate	4	5*6 = 30
5		Push (30)	4, 30	
6	+	Pop (2 element) and evaluate	Empty	4+30=34
7		Push (34)	34	
8		No-more elements (pop)	Empty	34 (Result)

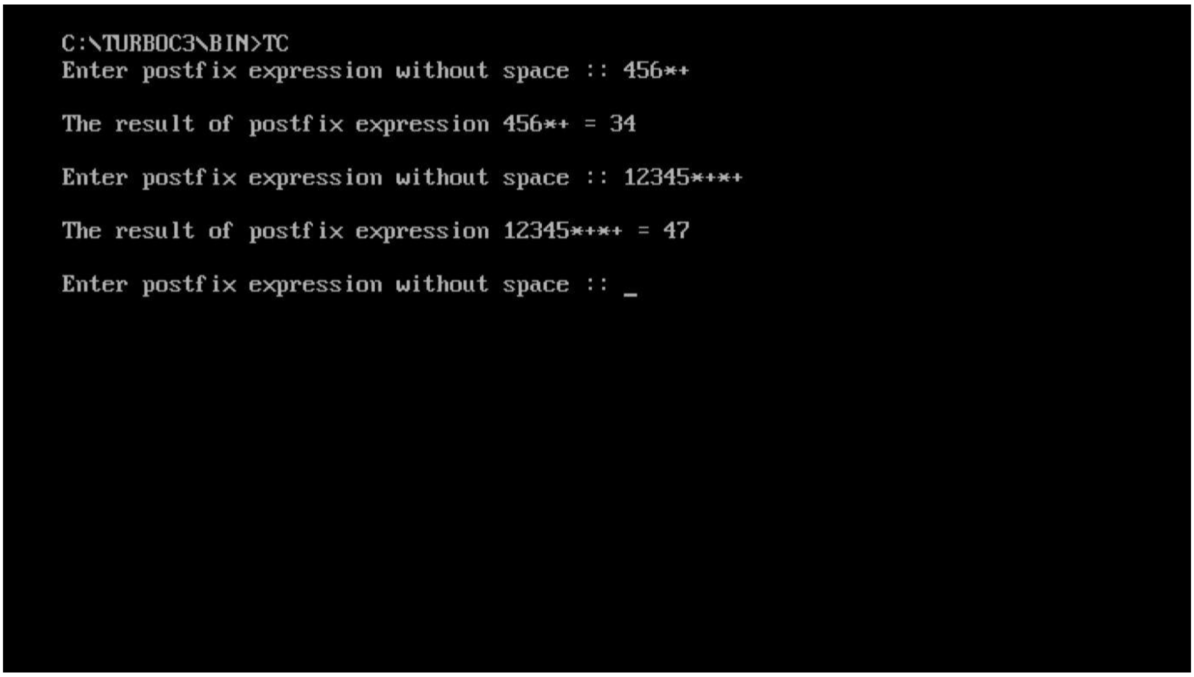
Step	Input	Operation	Stack	Calculation
1	1	Push	1	
2	2	Push	1, 2	
3	3	Push	1, 2, 3	
4	4	Push	1, 2, 3, 4	
5	5	Push	1, 2, 3, 4, 5	
6	*	Pop	1, 2, 3	4 * 5 = 20
7		Push (20)	1, 2, 3, 20	
8	+	Pop	1, 2	3+20=23
9		Push (23)	1, 2, 23	
10	*	Pop	1	2 * 23 = 46
11		Push (46)	1, 46	
12	+	Pop		1+46=47

Sundaram®
FOR EDUCATIONAL USE

PROGRAM:

```
#include<stdio.h>
int stack [20];
int top=-1;
void push(int x) {
stack [++top] = x;
}
int pop() {
return stack[top--];
}
void main() {
char exp[20], *temp;
int n1, n2, n3, num;
setbuf(stdout, NULL);
printf("Enter postfix expression without space :: ");
scanf("%s", exp);
temp = exp;
while(*temp != '\0') {
if(isdigit(*temp)) {
num = *temp - 48;
push (num);
}
else {
n1 = pop(); //operand1
n2 = pop(); //operand2
switch(*temp) {
case '+': {
n3 = n2 + n1;
break;
}
case '-': {
n3 = n2 - n1;
break;
}
case '*': {
n3 = n2 * n1;
break;
}
case '/' : {
n3 = n2 / n1;
break;
}
}
push(n3);
}
temp++;
}
printf("\nThe result of postfix expression %s = %d\n\n", exp, pop());
}
```

OUTPUT:-



```
C:\TURBOC3\BIN>TC
Enter postfix expression without space :: 456*+

The result of postfix expression 456*+ = 34

Enter postfix expression without space :: 12345*+*+

The result of postfix expression 12345*+*+ = 47

Enter postfix expression without space :: _
```