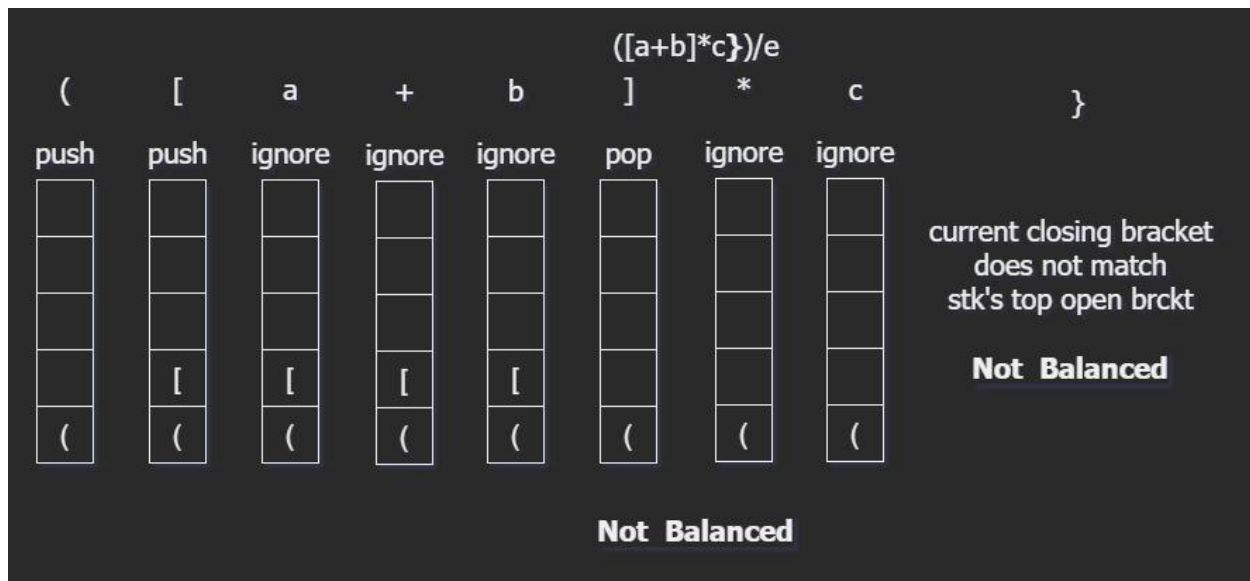Question 1   Ex.No: 4

AIM :

To check matching brackets using **Stack ADT**

DIAGRAMS :

([a+b]*c)/e]

(	[	a	+	b	]	*	c	)	/	e	]
push	push	ignore	ignore	ignore	pop	ignore	ignore	pop	ignore	ignore	

current closing bracket
has no
stk's top open brckt

Not Balanced

Stack contents:
- [[[[columns]
- (((((((columns]

Not Balanced

([a+b]*c)/e [

(	[	a	+	b	]	*	c	)	/	e	[
push	push	ignore	ignore	ignore	pop	ignore	ignore	pop	ignore	ignore	push

stk is not empty => **Not Balanced**
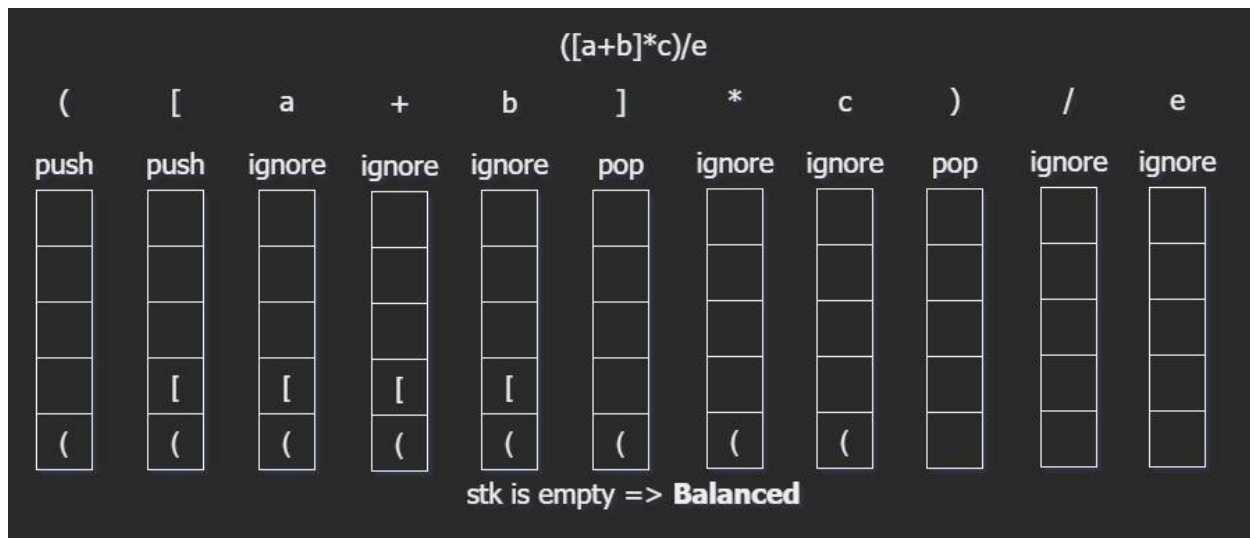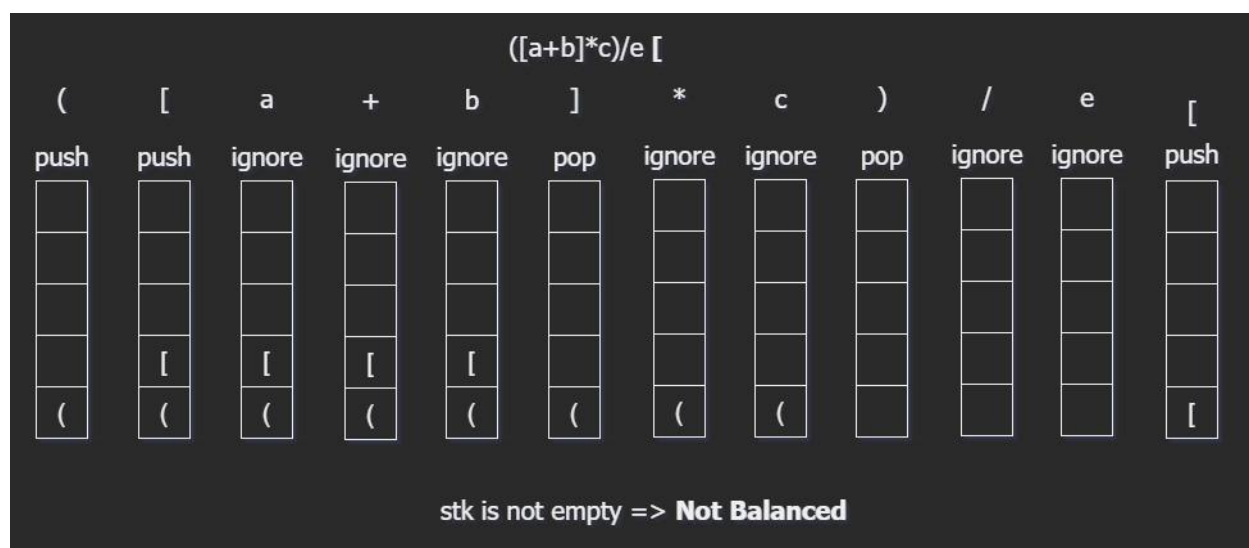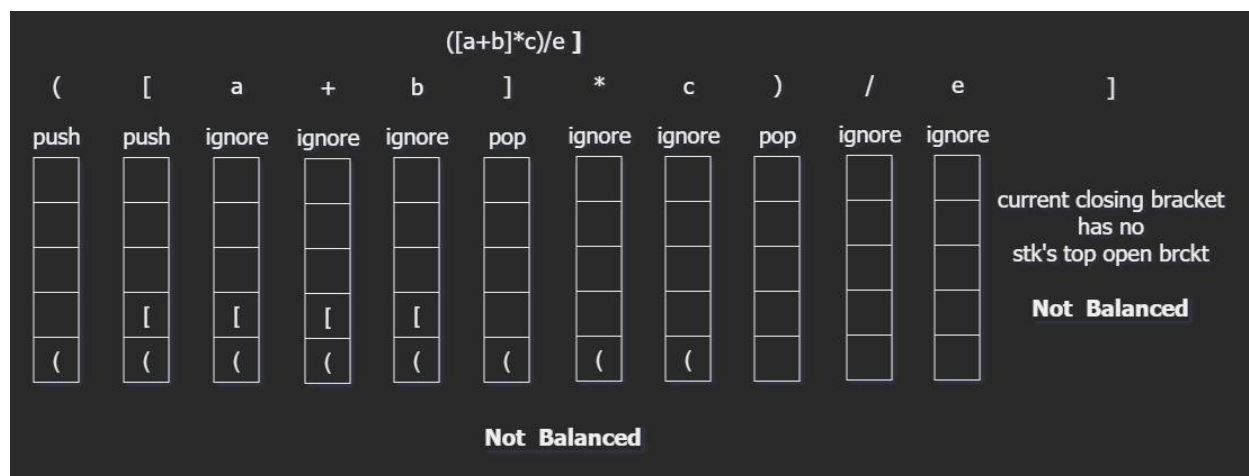
CODE :

```c
int isBalanced(char* exp) {
    struct Stck* stk = createStck();
    int i = 0;
    while (exp[i]) {
        if (exp[i] == '(' || exp[i] == '{' || exp[i] == '[')
            stkPush(stk, exp[i]);
        if (exp[i] == ')' || exp[i] == '}' || exp[i] == ']') {
            if ((stkTop(*stk) == '(' && exp[i] == ')') ||
                (stkTop(*stk) == '{' && exp[i] == '}') ||
                (stkTop(*stk) == '[') && exp[i] == ']')
                stkPop(stk);
            else return 0;
        }
        i++;
    }
    if (!isStkEmpty(*stk)) return 0;
    return 1;
}
```

```c
int ptBlnced(char* exp) {
    if (isBalanced(exp)) printf("%s is Balanced\n", exp);
    else printf("%s is not balanced\n", exp);
}
int main(int argc, char const* argv[]) {
    ptBlnced("([a+b]*c)/e");
    ptBlnced("{([a+b]}*[c-d])/e}");
    ptBlnced("{([{a+b]*[c-d])/e}");
    printf("\nProgram Terminated\n");
    return 0;
}
```

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

struct Node {
    struct Node* next;
    char ch;
};

struct Stck {
    struct Node* top;
};

struct Stck* createStck() {
    struct Stck* stk = (struct Stck*)(calloc(1, sizeof(struct Stck)));
    stk->top = NULL;
    return stk;
}

char stkTop(struct Stck stk) {
    if (!stk.top) return '\0';
    return stk.top->ch;
}



int isStkEmpty(struct Stck stk) {
    return stkTop(stk) == '\0';
}
```

```c
void stkPush(struct Stck* stk, char ch) {
    struct Node* newNode = (struct Node*)(calloc(1, sizeof(struct Node)));
    newNode->ch = ch;
    newNode->next = stk->top;
    stk->top = newNode;
}

char stkPop(struct Stck* stk) {
    if (!stk->top) return '\0';
    struct Node* nd = stk->top;
    char c = stk->top->ch;
    stk->top = stk->top->next;
    free(nd);
    return c;
}
```

OUTPUT :

```
D:\Shreeram\A_SEM3\DS\E4>mb.exe
{([a+b]*[c-d])/e} is Balanced
{([a+b]}*[c-d])/e} is not balanced
{([{a+b]*[c-d])/e} is not balanced

Program Terminated

D:\Shreeram\A_SEM3\DS\E4>
```