

PRACTICAL NO : 5-b

AIM : stack implementation using linked list

Program :

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

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

```
struct Node {  
    int data;  
    struct Node* next;  
};
```

```
struct Stack {  
    struct Node* top;  
};
```

```
void initialize(struct Stack* stack) {  
    stack->top = NULL;  
}
```

```
void push(struct Stack* stack, int value) {  
    struct Node* newNode = (struct Node*)malloc(sizeof(struct  
Node));
```

```

if (!newNode) {
    printf("Memory allocation failed!\n");
    return;
}
newNode->data = value;
newNode->next = stack->top;
stack->top = newNode;
printf("%d pushed to stack.\n", value);
}

void pop(struct Stack* stack) {
    if (stack->top == NULL) {
        printf("Stack is empty! Cannot pop.\n");
        return;
    }
    struct Node* temp = stack->top;
    stack->top = stack->top->next;
    printf("%d popped from stack.\n", temp->data);
    free(temp);
}

```

```

void printStack(struct Stack* stack) {
    if (stack->top == NULL) {
        printf("Stack is empty.\n");
        return;
    }
}

```

```

    }
    struct Node* temp = stack->top;
    printf("Stack elements: ");
    while (temp != NULL) {
        printf("%d ", temp->data);
        temp = temp->next;
    }
    printf("\n");
}

```

```

void freeStack(struct Stack* stack) {
    while (stack->top != NULL) {
        pop(stack);
    }
}

```

```

int main() {
    struct Stack stack;
    initialize(&stack);

    int choice, value;

    while (1) {
        printf("\nMenu:\n");
        printf("1. Push\n");

```

```
printf("2. Pop\n");
printf("3. Print\n");
printf("4. Exit\n");
printf("Enter your choice: ");
scanf("%d", &choice);

switch (choice) {
    case 1:
        printf("Enter value to push: ");
        scanf("%d", &value);
        push(&stack, value);
        break;

    case 2:
        pop(&stack);
        break;

    case 3:
        printStack(&stack);
        break;

    case 4:
        freeStack(&stack);
        printf("Exiting...\n");
```

```
exit(0);
```

```
default:
```

```
printf("Invalid choice! Try again.\n");
```

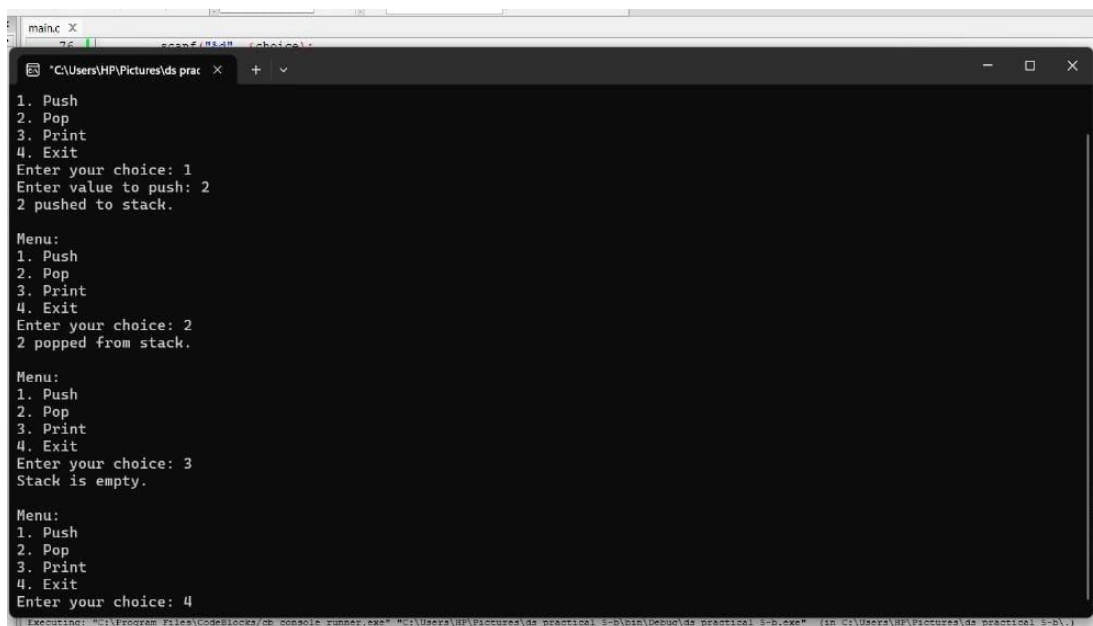
```
}
```

```
}
```

```
return 0;
```

```
}
```

OUTPUT :



```
main.c X
C:\Users\HP\Pictures\ds prac
1. Push
2. Pop
3. Print
4. Exit
Enter your choice: 1
Enter value to push: 2
2 pushed to stack.

Menu:
1. Push
2. Pop
3. Print
4. Exit
Enter your choice: 2
2 popped from stack.

Menu:
1. Push
2. Pop
3. Print
4. Exit
Enter your choice: 3
Stack is empty.

Menu:
1. Push
2. Pop
3. Print
4. Exit
Enter your choice: 4
```

