Zayd Ahmed

1BM21CS254

Q: Stack Implementation using Pointers

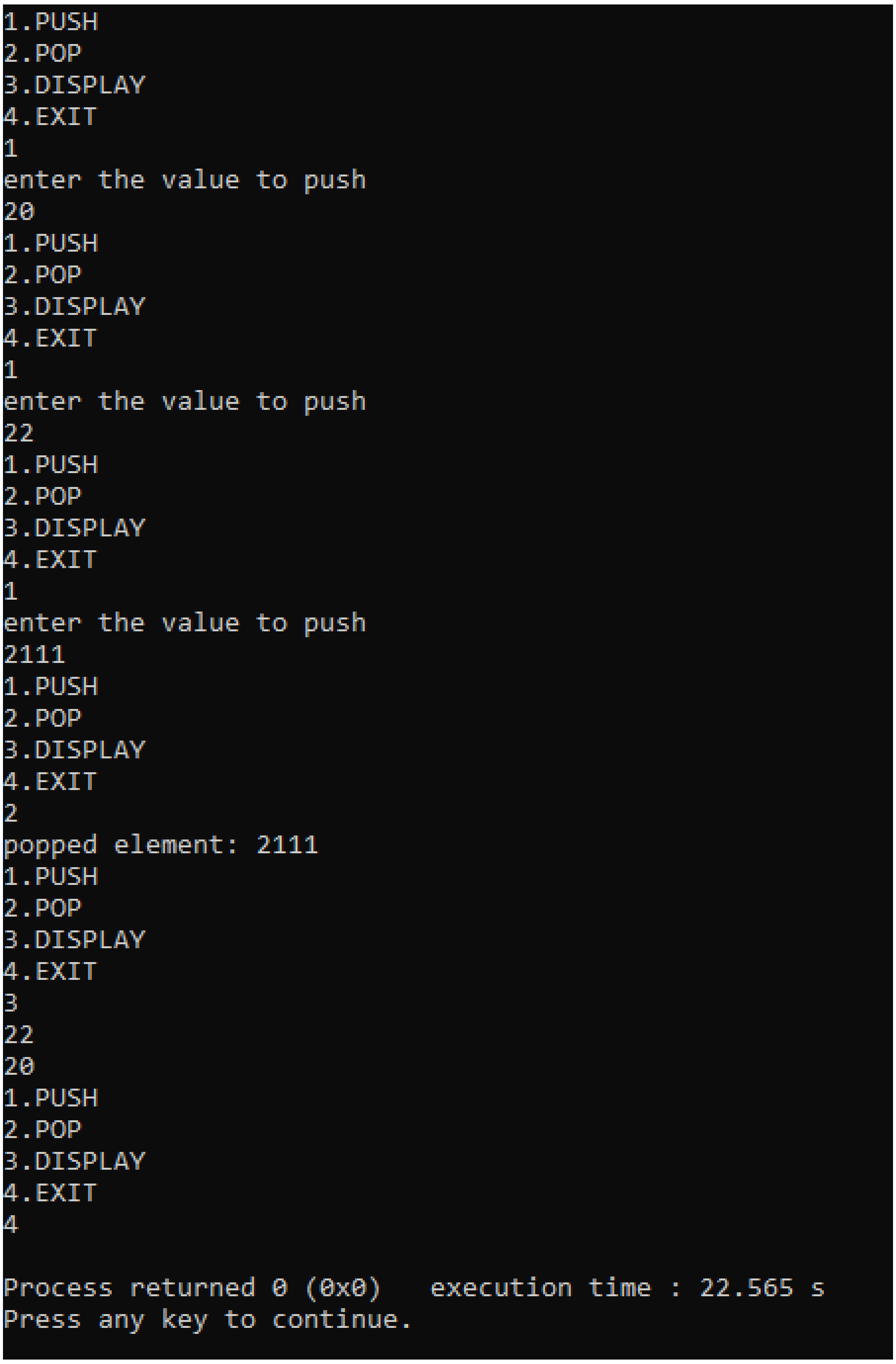
Code:-

```
#include <stdio.h>
#include<stdlib.h>
#define size 3

struct stack{
   int s[size];
   int top;
};
void push(struct stack *p)
{
   if((p->top)>=(size-1)){
      printf("stack overflow\n");
   }
```

```
    else{
        printf("enter the value to push\n");
        int n;
        scanf("%d",&n);
        p->top++;
        p->s[p->top]=n;
    }
};
int pop(struct stack *p)
{
    if(p->top==-1){
        printf("stack underflow\n");
        return;
    }
    else{
        int n;
        n=p->s[p->top];
        p->top--;
        return n;
    }
```

```
};
void display(struct stack *p){
    if(p->top==-1){
        printf("empty stack\n");
        return;
    }
    else{
        for(int i=(p->top);i>-1;i--){
            printf("%d\n",p->s[i]);
        }
    }
};

int main(){
    struct stack st;
    st.top=-1;
    int choice;
    int del;
    while(1){
        printf("1.PUSH\n2.POP\n3.DISPLAY\n4.EXIT\n");
        scanf("%d",&choice);

        switch(choice){

            case 1: push(&st);

                    break;

            case 2: del=pop(&st);

                    printf("popped element: %d\n",del);

                    break;

            case 3: display(&st);

                    break;

            case 4: exit(0);

            default:printf("enter a valid choice\n");

        }

    }

    return 0;

}
```

Output:-



```
1.PUSH
2.POP
3.DISPLAY
4.EXIT
1
enter the value to push
20
1.PUSH
2.POP
3.DISPLAY
4.EXIT
1
enter the value to push
22
1.PUSH
2.POP
3.DISPLAY
4.EXIT
1
enter the value to push
2111
1.PUSH
2.POP
3.DISPLAY
4.EXIT
2
popped element: 2111
1.PUSH
2.POP
3.DISPLAY
4.EXIT
3
22
20
1.PUSH
2.POP
3.DISPLAY
4.EXIT
4

Process returned 0 (0x0)   execution time : 22.565 s
Press any key to continue.
```