

(a)

push, pop, display.

```
int top = -1, stack[1000];
```

```
void push()
```

```
void pop()
```

```
void display()
```

```
void main()
```

```
{
```

```
    int ch;
```

```
    while(1)
```

```
    {
```

```
        printf("\n\n 1. Push\n 2. Pop\n 3. Display\n");
```

```
        printf("Enter your choice");
```

```
        scanf("%d", &ch);
```

```
        switch(ch)
```

```
        {
```

```
            case 1: push();
```

```
                break;
```

```
            case 2: pop();
```

```
                break;
```

```
            case 3: display();
```

```
                break;
```

```
            case 4: exit(0);
```

```
            default: printf("Wrong choice");
```

```
        }
```

```
    }
```

```
void push()
```

```
{
```

```
    int val;
```

```
    if (top == max - 1)
```

```
    {
```

```
        printf("\n stack is full!");
```

```
    }
```

```
    else
```

```
    {
```

```
        printf("Enter element to push:");
```

```
        scanf("%d", &val);
```

```
        top = top + 1;
```

```
        stack[top] = val;
```

```
    }
```

```
}
```

```
void pop()
```

```
{
```

```
    if (top == -1)
```

```
    {
```

```
        printf("\n stack is empty!!");
```

```
    }
```

```
    else
```

```
    {
```

```
        printf("\n deleted element is %d", stack[top]);
```

```
        top = top - 1;
```

```
    }
```

```
}
```

```
void display()
```

```
{
```

```
    int i;
```

```
    if (top == -1)
```

```
    {
```

```
        printf("Stack is empty\n");
```

```
    }
```

else

{

printf ("Stack is --- \n");

for (i = top; i >= 0; i--)

printf ("%d \n", stack[i]);

}

}