

Code:

```
#include <stdio.h>

#include <string.h>


#define MAX 5


int stack[MAX], top = -1;


int push(int x) {
    if (top == MAX - 1)
        return 0; // overflow → invalid
    stack[++top] = x;
    return 1;    // valid
}


int pop() {
    if (top == -1)
        return 0; // underflow → invalid
    top--;
    return 1;    // valid
}


int main() {
    int n, i, op, val;
    int valid = 1;

    printf("Enter number of operations: ");
    scanf("%d", &n);

    printf("\nEnter operations:\n");
```

```
printf("1 x → push x\n");
```

```
printf("2 → pop\n\n");
```

```
for (i = 0; i < n; i++) {
```

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

```
    if (op == 1) {
```

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

```
        if (!push(val)) {
```

```
            valid = 0;
```

```
            break;
```

```
        }
```

```
    } else if (op == 2) {
```

```
        if (!pop()) {
```

```
            valid = 0;
```

```
            break;
```

```
        }
```

```
    } else {
```

```
        valid = 0;
```

```
        break;
```

```
    }
```

```
}
```

```
if (valid)
```

```
    printf("\nVALID STACK SEQUENCE\n");
```

```
else
```

```
    printf("\nINVALID STACK SEQUENCE\n");
```

```
return 0;
```

```
}
```

OUTPUT:

```
Enter number of operations: 6
```

```
Enter operations:
```

```
1 x → push x
```

```
2   → pop
```

```
1
```

```
2
```

```
3
```

```
INVALID STACK SEQUENCE
```