

```

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

typedef struct {
    int top; //indice dell'elemento affiorante
    int* A; //array dinamico degli elementi
    int A_length; // Lunghezza di A: E' una tecnica
} Stack;

/**
 * @brief Nuovo Staack
 * @return
 */
Stack* new_stack(){
    Stack* s = (Stack*) malloc(sizeof(Stack));
    s->top = -1;
    s->A = (int*) calloc(16, sizeof(int));
    s->A_length = 16;
    return s;
}

/**
 * @brief Controllo se la coda è vuota
 * @param s
 * @return
 */
int IS_EMPTY(Stack* s){
    return s->top == -1;
}

/**
 * @brief Inserimento
 * @param s
 * @param x
 */
void PUSH(Stack* s, int x){
    if(s->top == s->A_length-1){ //Array pieno
        s->A_length = 2 * s->A_length;
        s->A = (int *) realloc(s->A, s->A_length*sizeof(int));
    }

    s->top = s->top + 1;
    s->A[s->top] = x;
}

int POP(Stack* s){
    if(IS_EMPTY(s)){
        printf("\nErrore: POP su stack vuoto.\n");
        exit(1);
    }

    s->top = s->top-1;
    return s->A[s->top + 1];
}

```

```
}

int main(int argc, char **argv)
{
    int i;
    Stack *s = new_stack();

    printf("\n\n");
    for(i = 1; i < 11; i++){
        printf("ora inserisco %d\n", i);
        PUSH(s, i);
    }

    printf("\n\n");
    for(i = 1; i < 11; i++){
        printf("ora estraggo %d\n", POP(s));
    }
    return 0;
}
```