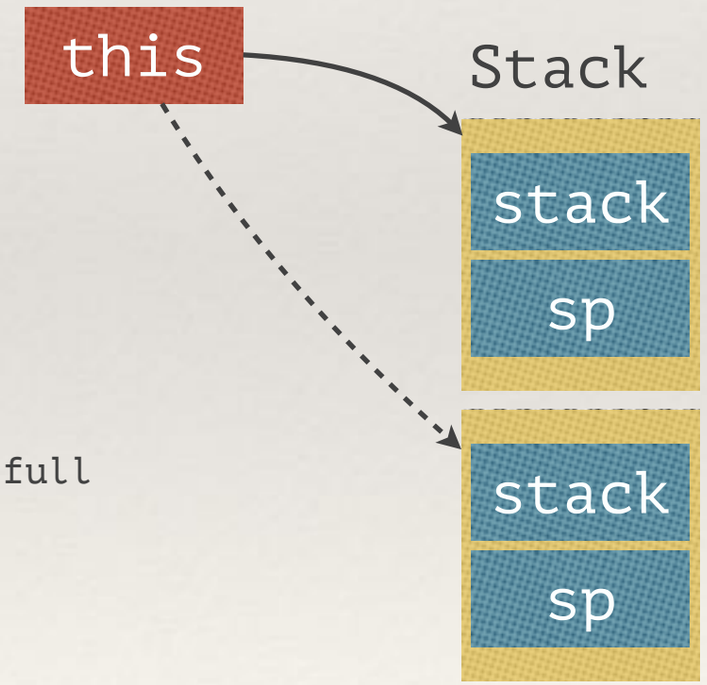
Sa proceduralnog na OO programiranje: klase i objekti

- Nedostatak ovog rešenja: u programu imamo samo jedan ovakav stek, jednu instancu.
  Šta ako nam je potrebno više instanci ovakve strukture? Organizacija koja bi ovo omogućila:

```
/* File: stack.h */

#define MaxStackSize 256

struct Stack {
  unsigned stack[MaxStackSize];  // Stack
  int sp; // Stack pointer
};

void stack_init (Stack* this);
int stack_push (Stack* this, unsigned in);
int stack_pop (Stack* this, unsigned* out);

/* File stack.c */
#include “stack.h”

void stack_init (Stack* this) {
  this->sp = 0;
}

int stack_push (Stack* this, unsigned in) {
  if (this->sp==MaxStackSize) return -1; // Exception: stack full
  this->stack[this->sp++] = in;
  return 0;
}

int stack_pop (Stack* this, unsigned* out) {
  if (this->sp==0) return -1; // Exception: stack empty
  *out = this->stack[--this->sp];
  return 0;
}
```



Sa proceduralnog na OO programiranje: klase i objekti

- Sada se ovo koristi ovako:

```
#include "stack.h"

Stack* pSt1 = …;
stack_init(pSt1);
…
unsigned out;
…
stack_push(pSt1,in);
…
stack_pop(pSt1,&out);
…
Stack* pSt2 = …;
stack_init(pSt2);
…
stack_push(pSt2,in);
…
stack_pop(pSt2,&out);
…
```