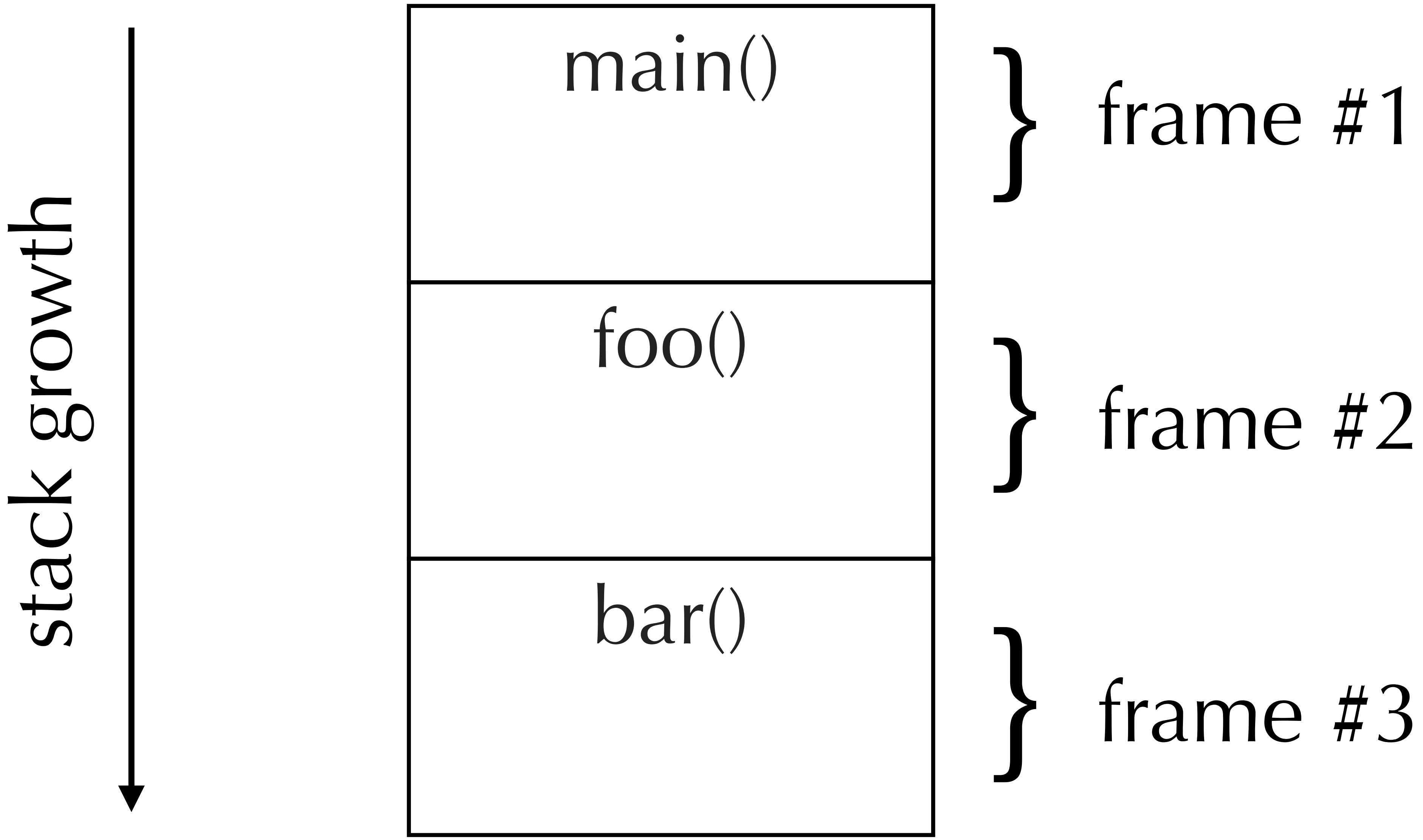
Coroutines

@danluu | github.com/danluu
@majek04 | github.com/majek
github.com/happy4crazy

Kinda like functions, but cooler

Applications

- Networking (gevent)
- OS (cooperative multi-tasking)
- Generators
- Foundation for higher-level concurrency models



stack growth

main()

foo()

bar()

frame #1

frame #2

frame #3

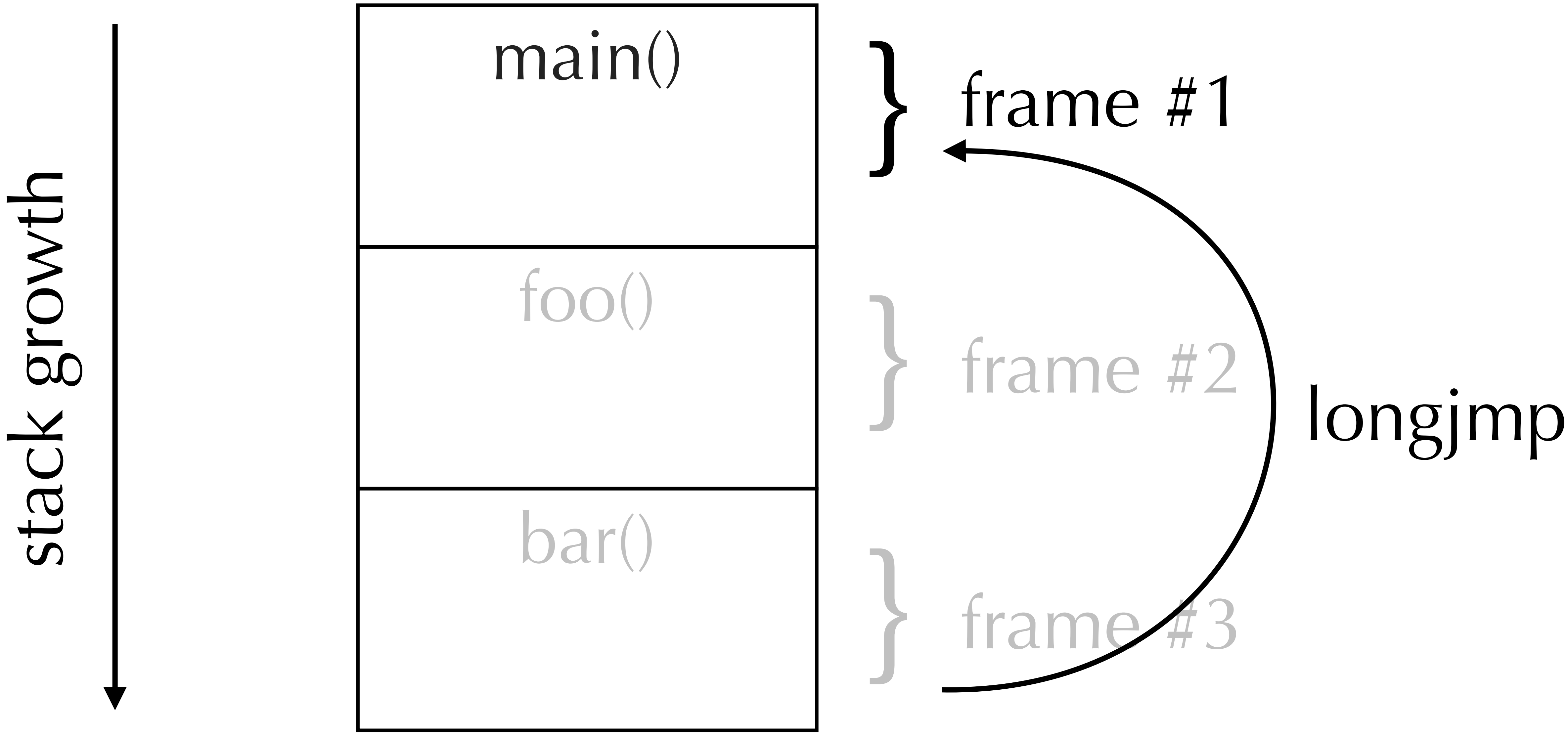
setjmp & longjmp

```
jmp_buf buf;

void bar() {
    longjmp(b);
}

void foo() {
    bar();
}

int main() {
    setjmp(b);
    foo();
}
```

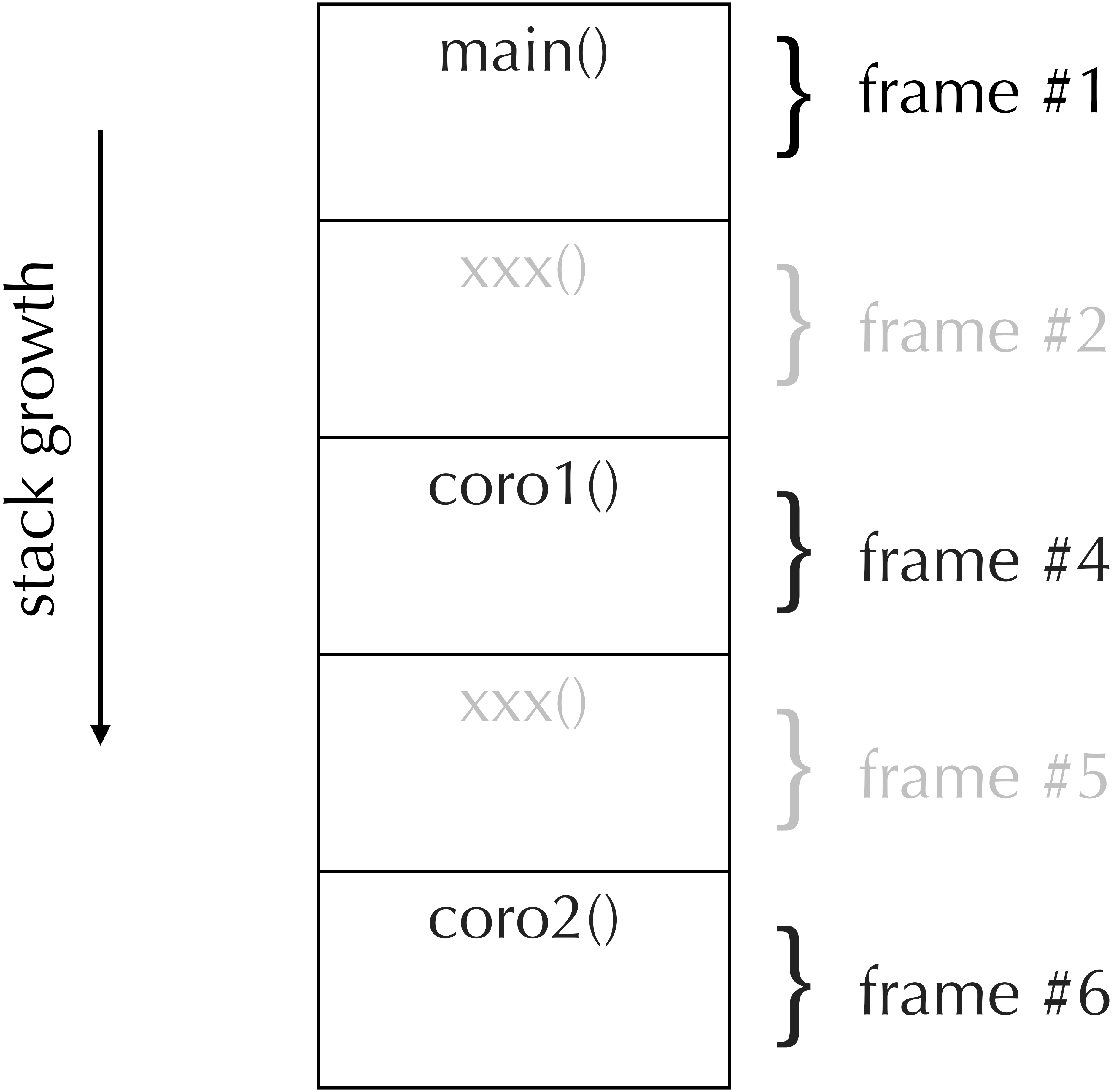

stack growth
main()
foo()
bar()
frame #1
frame #2
frame #3
longjmp

```c
jmp_buf buf;

void bar() {
    longjmp(b);
}

void foo() {
    bar();
}

int main() {
    setjmp(b);
    foo();
}
```



stack growth

main()

frame #1

xxx()

frame #2

coro1()

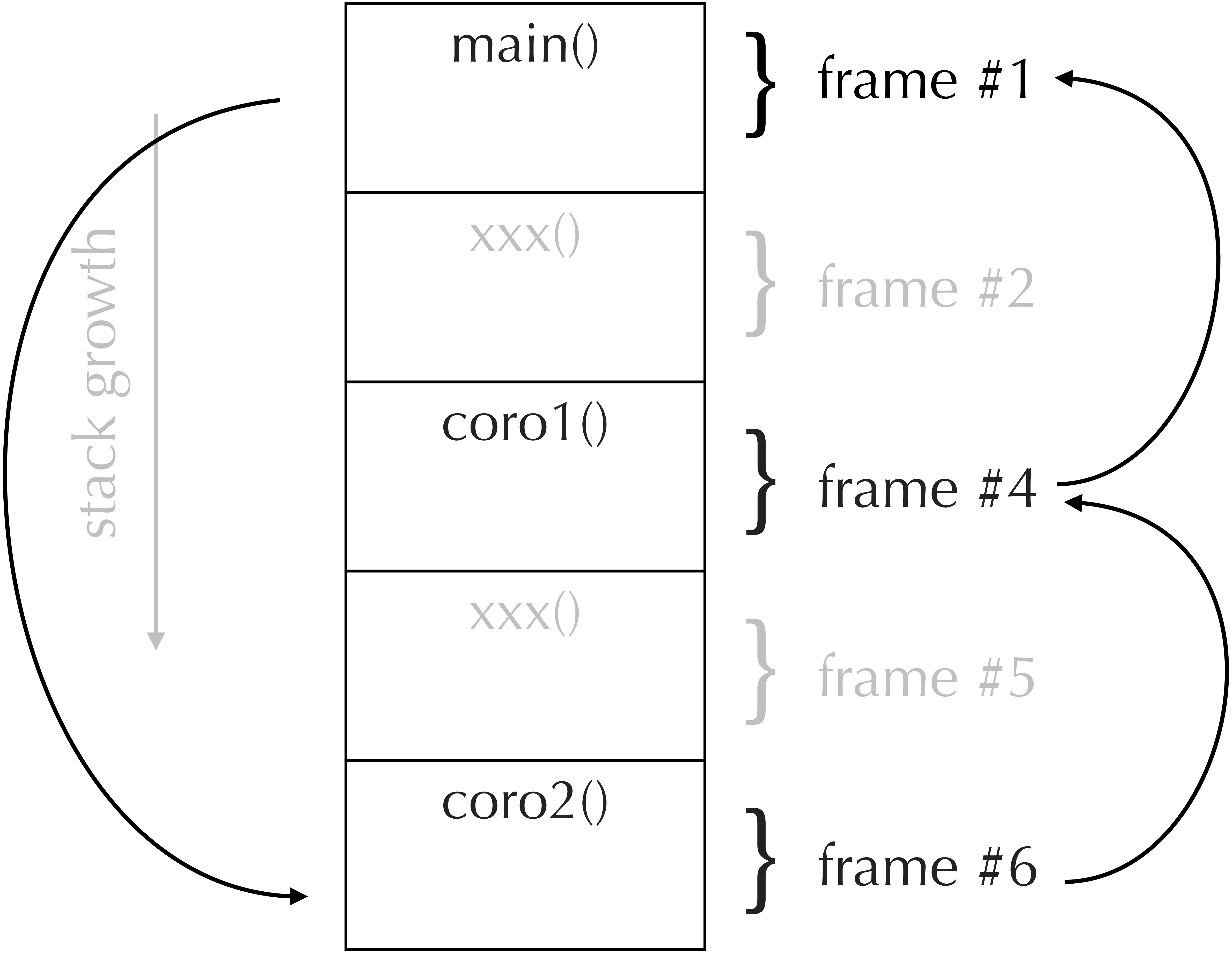
frame #4

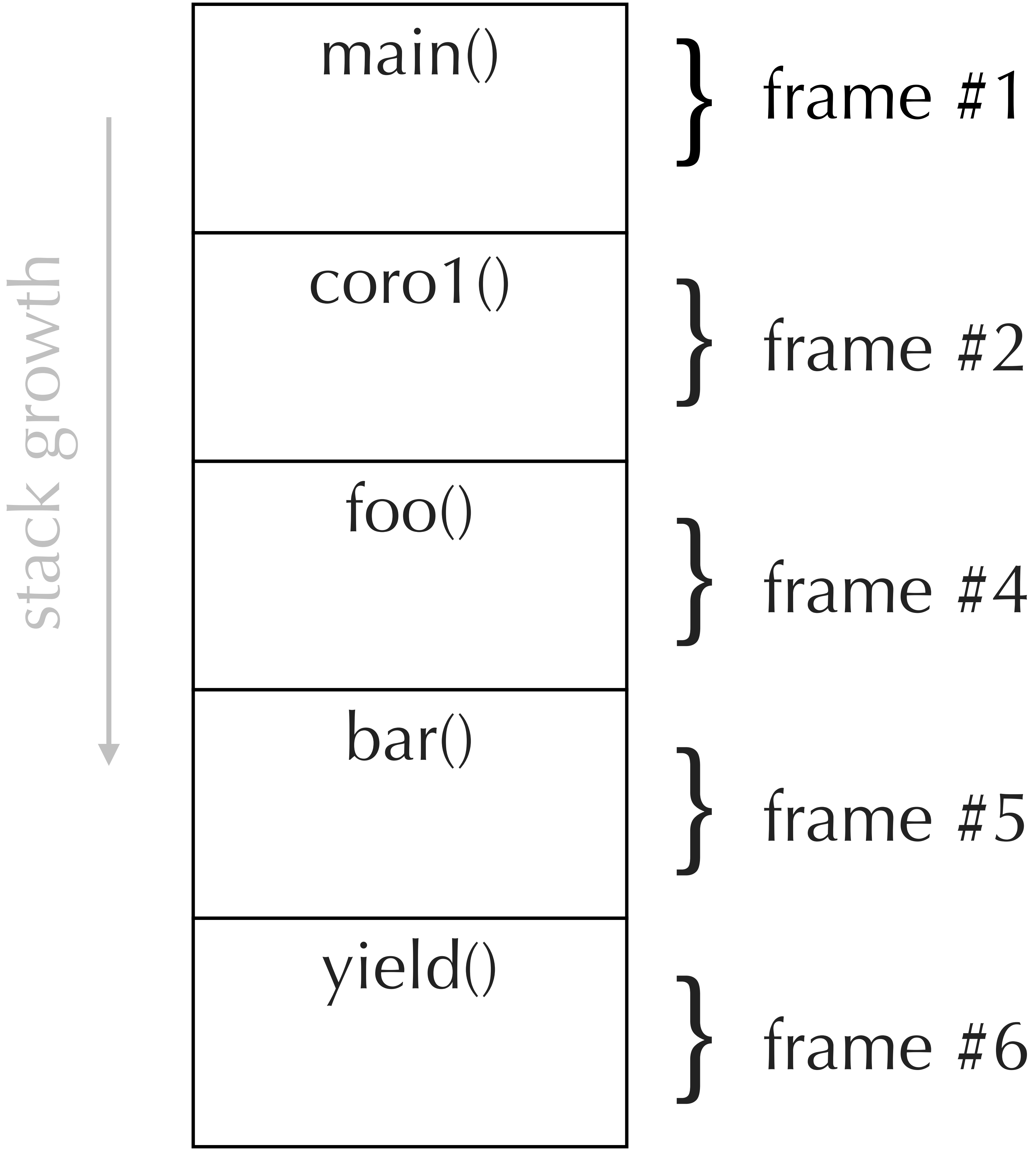
xxx()

frame #5

coro2()

frame #6


main()
xxx()
coro1()
xxx()
coro2()
stack growth
frame #1
frame #2
frame #4
frame #5
frame #6



stack growth

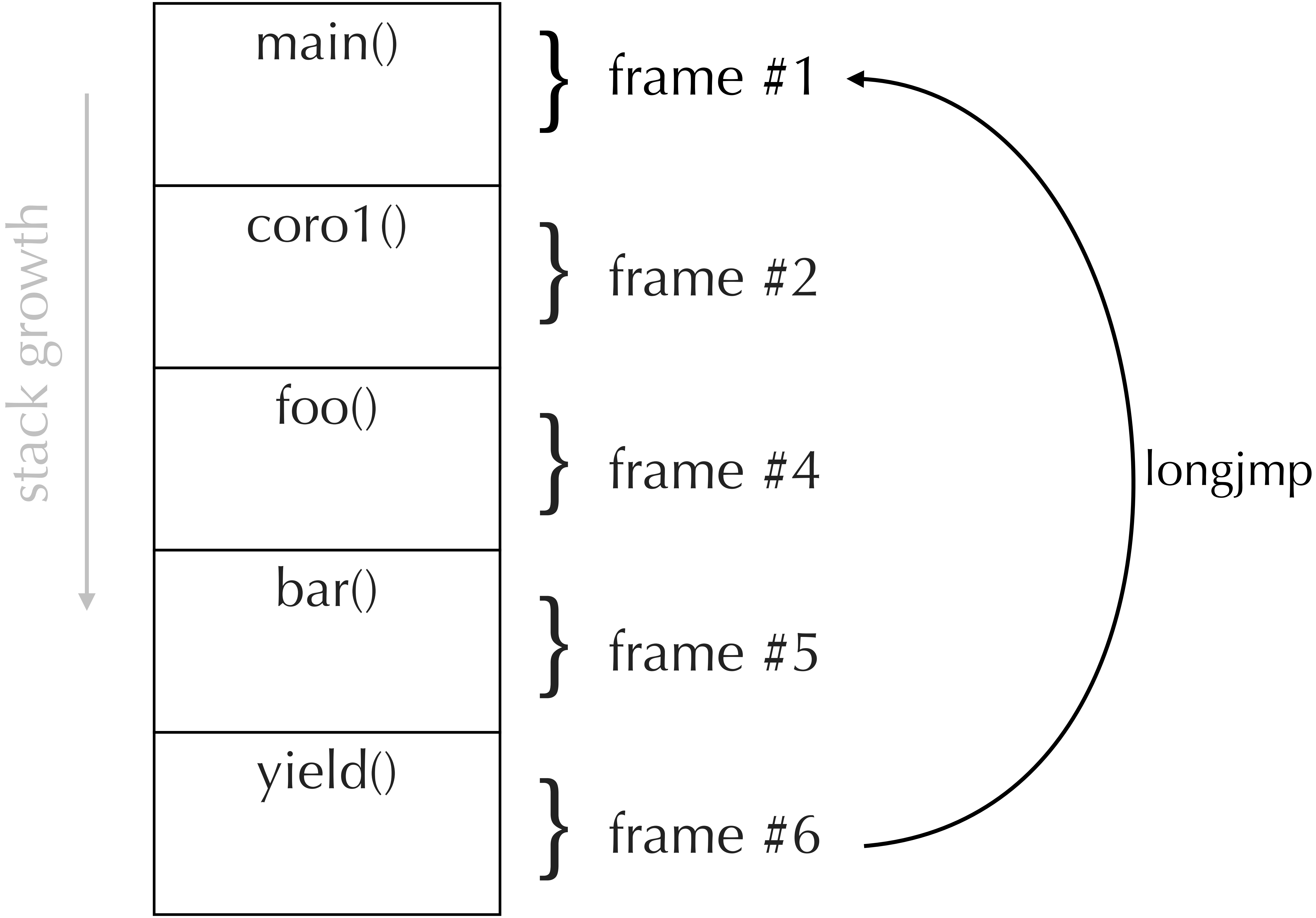
main() } frame #1

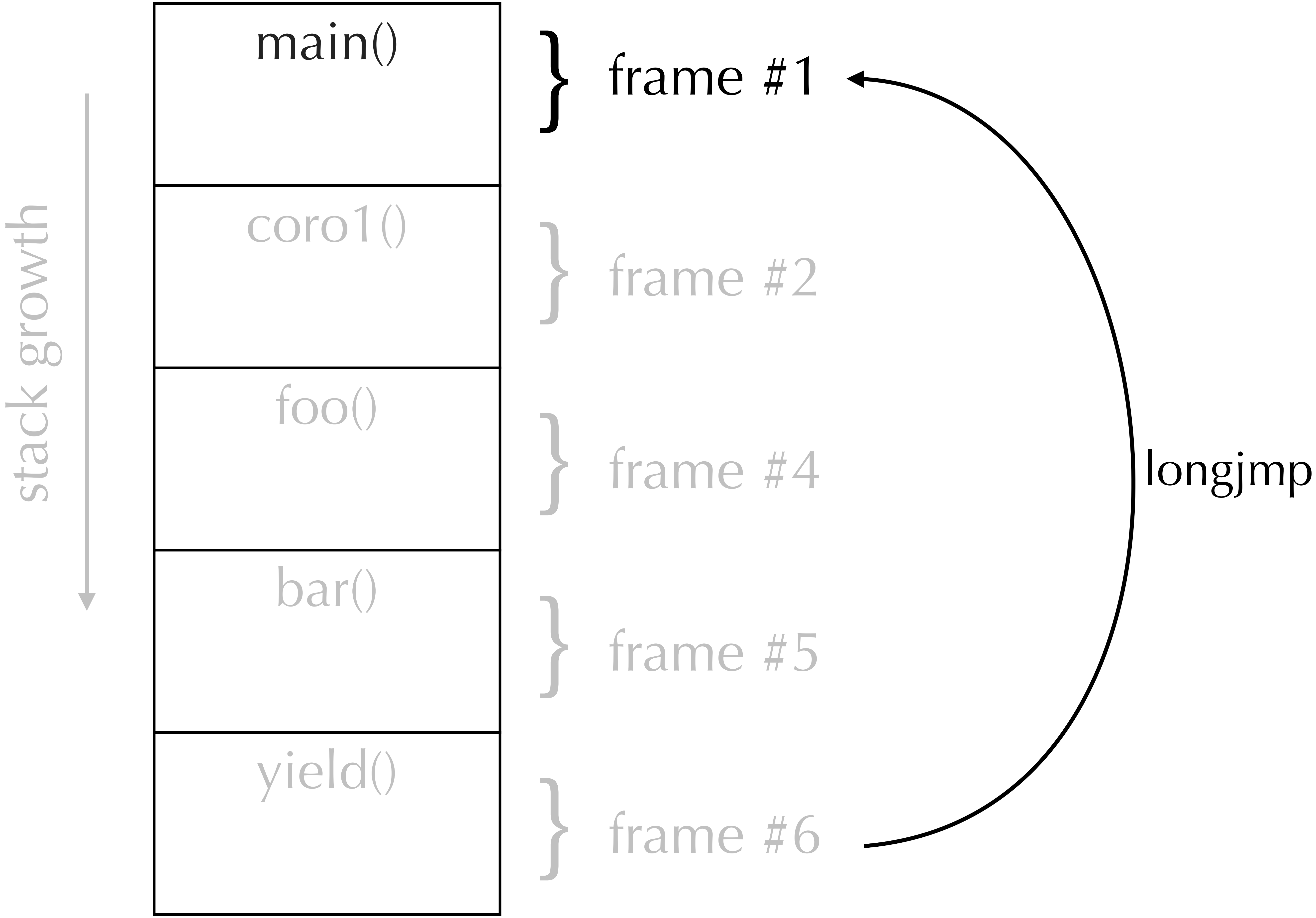
coro1() } frame #2

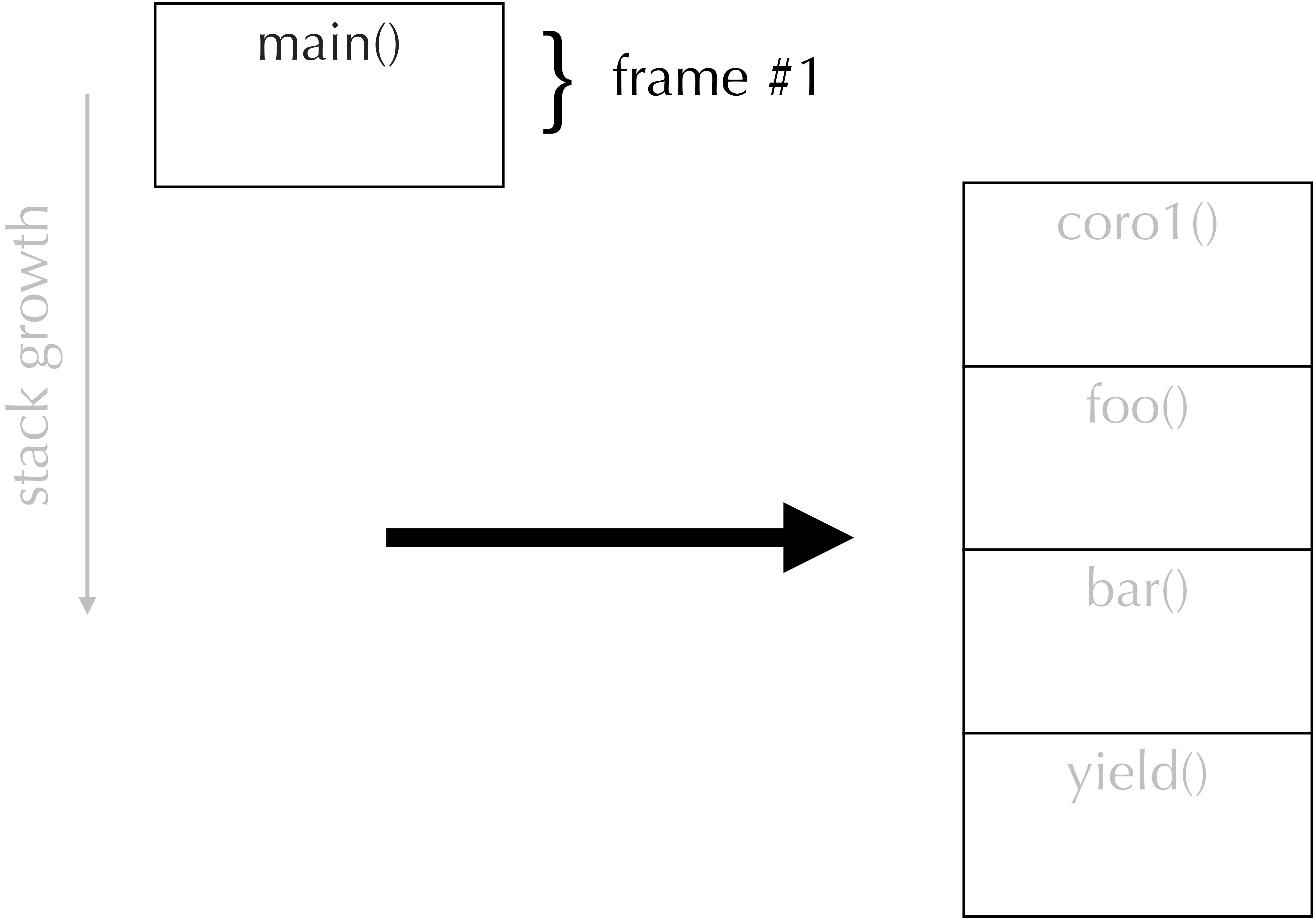
foo() } frame #4

bar() } frame #5

yield() } frame #6


stack growth
main()
coro1()
foo()
bar()
yield()
frame #1
frame #2
frame #4
frame #5
frame #6
longjmp


main()
frame #1
stack growth
coro1()
frame #2
foo()
frame #4
longjmp
bar()
frame #5
yield()
frame #6



main()

frame #1

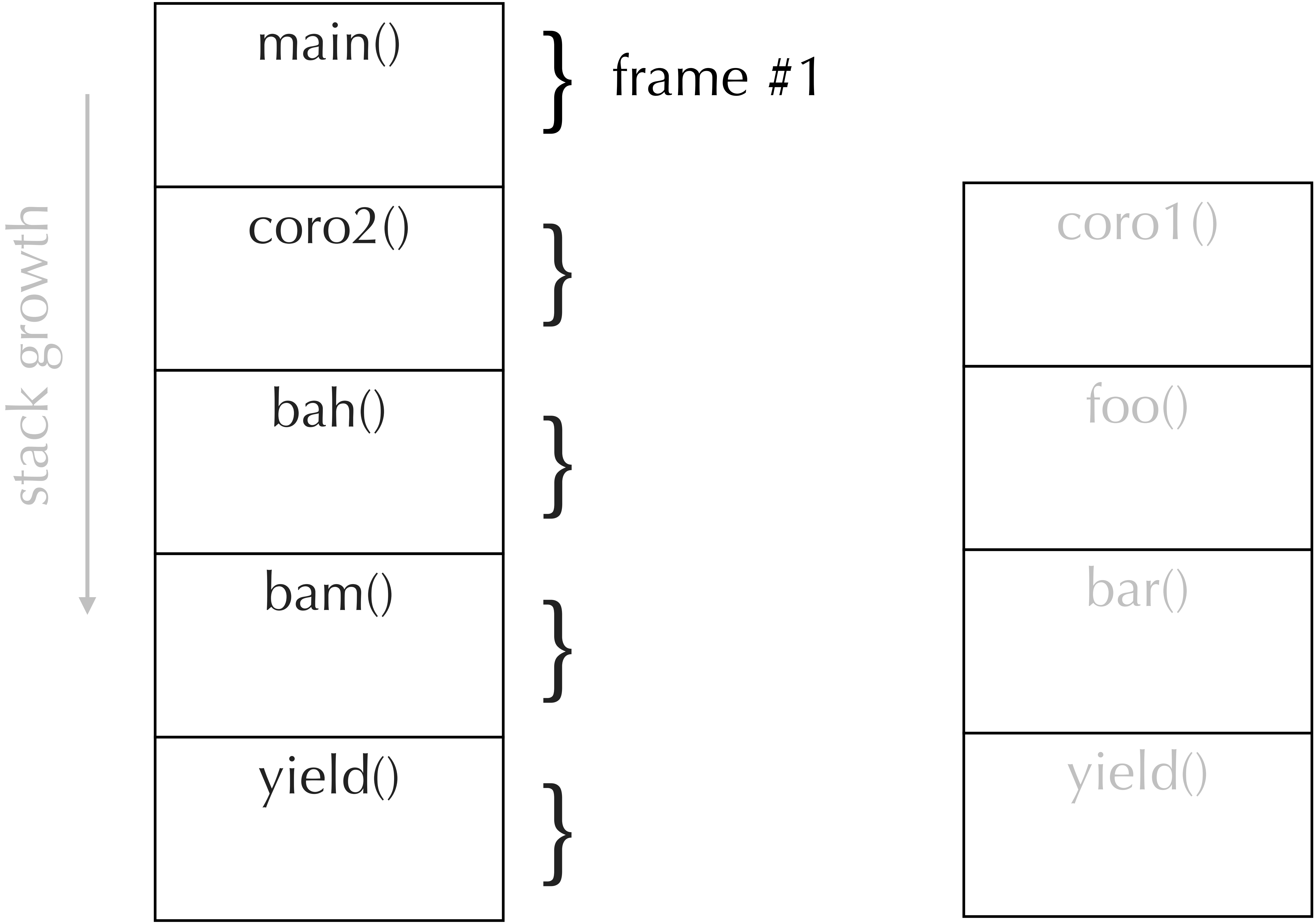
stack growth

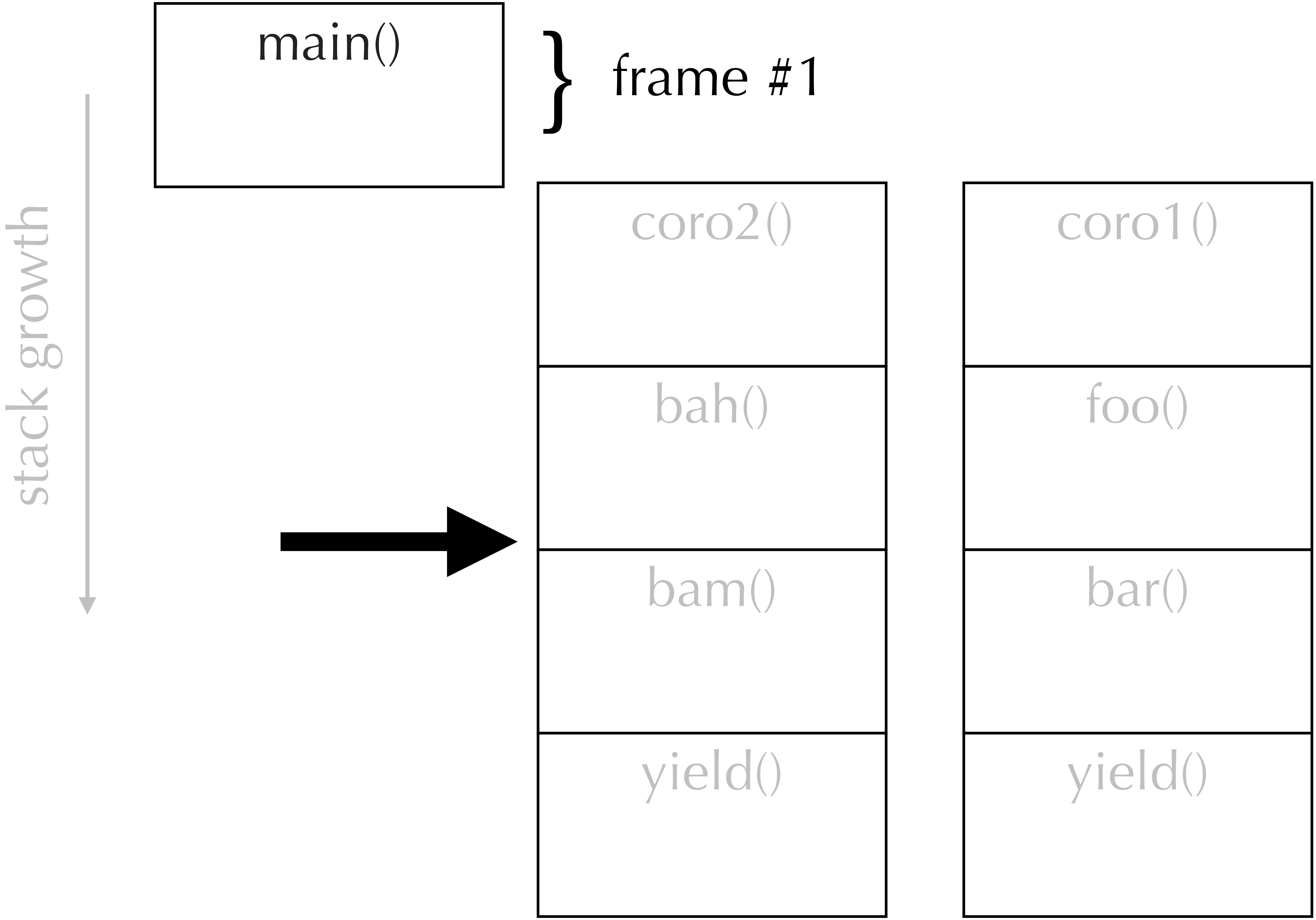
coro1()

foo()

bar()

yield()


stack growth
main()
frame #1
coro2()
bah()
bam()
yield()
coro1()
foo()
bar()
yield()



main()

frame #1

stack growth

coro2()

bah()

bam()

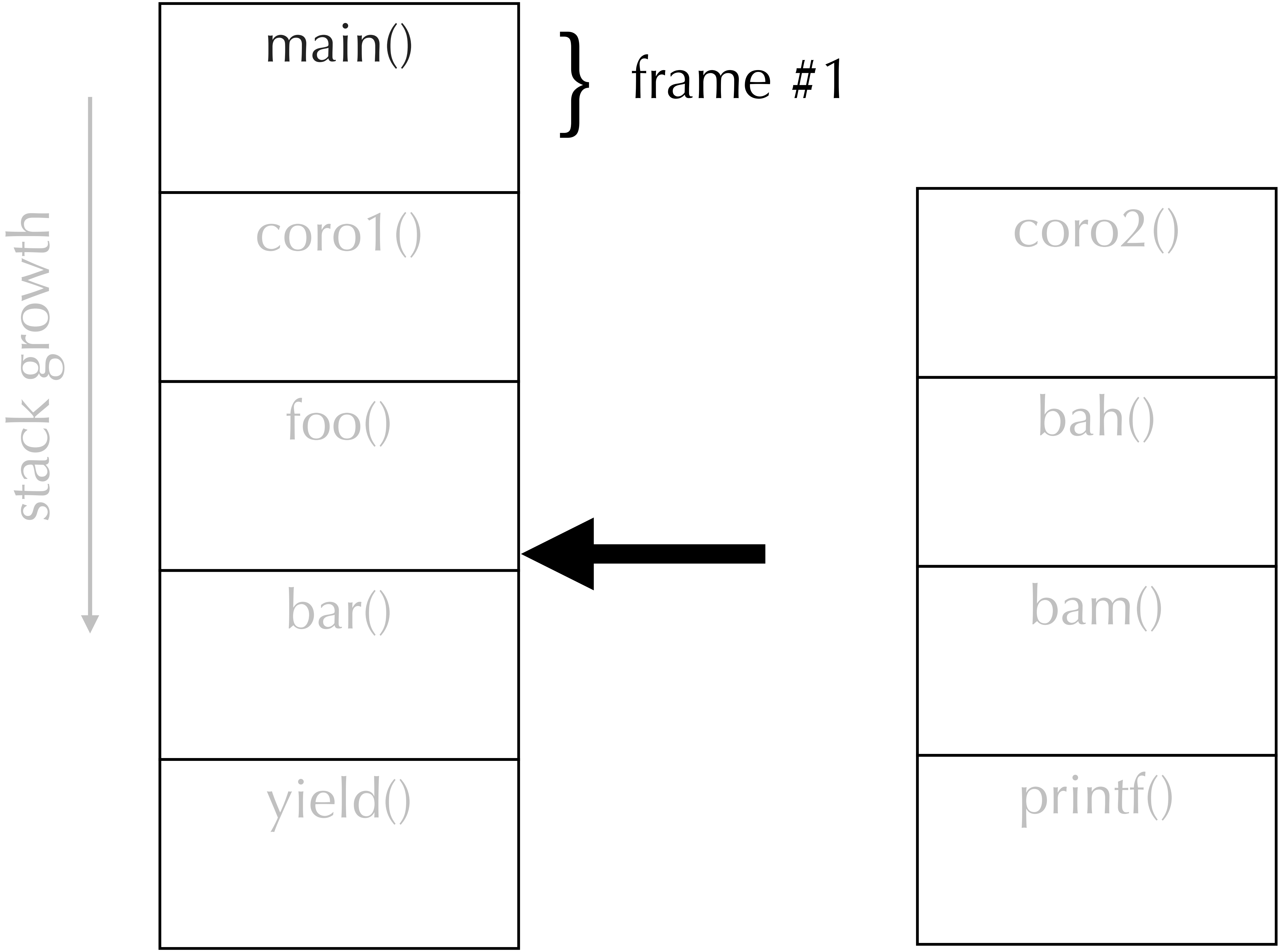
yield()

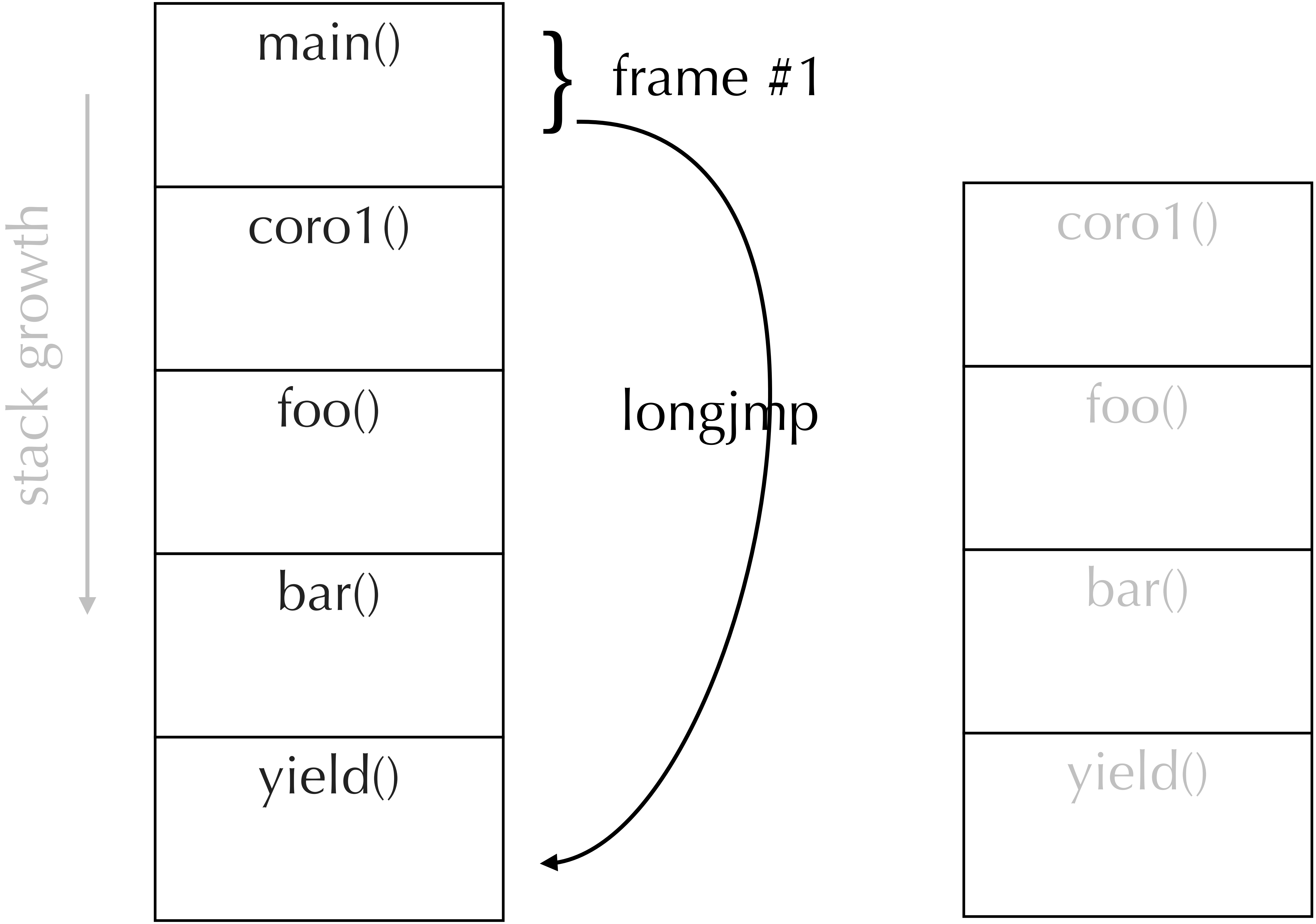
coro1()

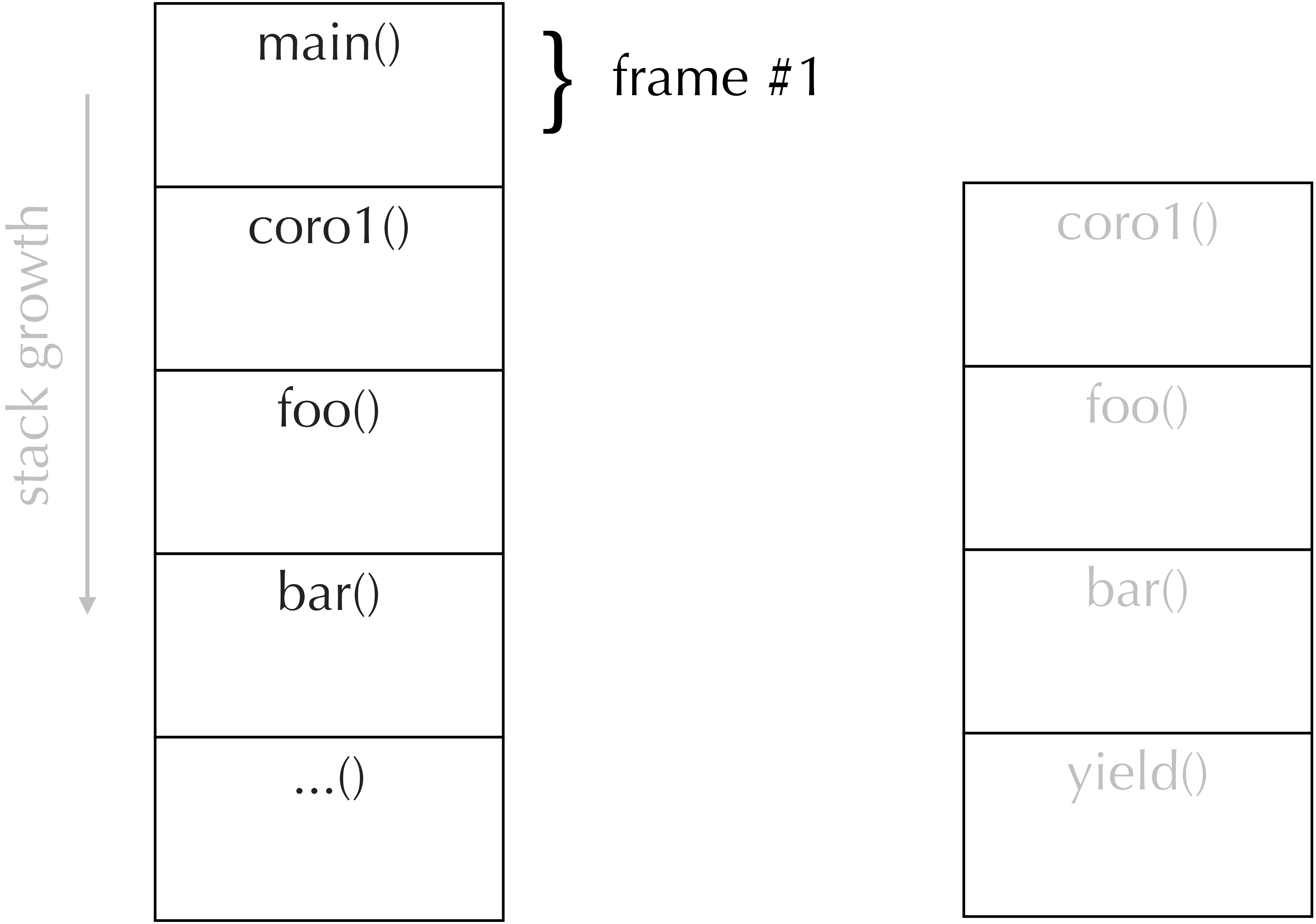
foo()

bar()

yield()


main()
frame #1
stack growth
coro1()
foo()
bar()
yield()
coro2()
bah()
bam()
printf()


stack growth
main()
coro1()
foo()
bar()
yield()
frame #1
longjmp
coro1()
foo()
bar()
yield()



stack growth

main()

frame #1

coro1()

foo()

bar()

...()

coro1()

foo()

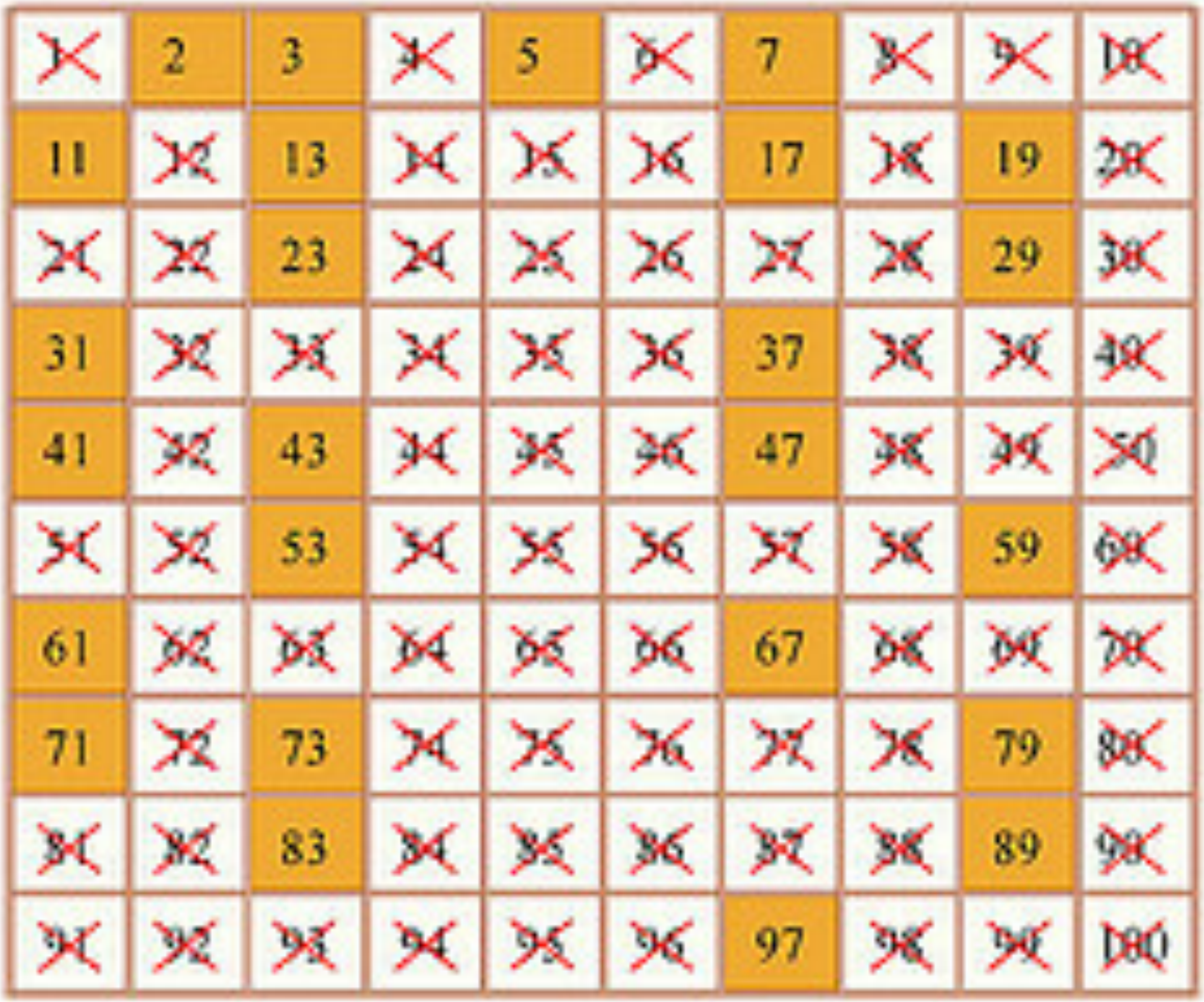
bar()

yield()

Channels

Go-style channels

- channel.send(value): blocking
- channel.recv(): blocking
- unblocks when someone on the other end



~~1~~	2	3	~~4~~	5	~~6~~	7	~~8~~	~~9~~	~~10~~
11	~~12~~	13	~~14~~	~~15~~	~~16~~	17	~~18~~	19	~~20~~
~~21~~	~~22~~	23	~~24~~	~~25~~	~~26~~	~~27~~	~~28~~	29	~~30~~
31	~~32~~	~~33~~	~~34~~	~~35~~	~~36~~	37	~~38~~	~~39~~	~~40~~
41	~~42~~	43	~~44~~	~~45~~	~~46~~	47	~~48~~	~~49~~	~~50~~
~~51~~	~~52~~	53	~~54~~	~~55~~	~~56~~	~~57~~	~~58~~	59	~~60~~
61	~~62~~	~~63~~	~~64~~	~~65~~	~~66~~	67	~~68~~	~~69~~	~~70~~
71	~~72~~	73	~~74~~	~~75~~	~~76~~	~~77~~	~~78~~	79	~~80~~
~~81~~	~~82~~	83	~~84~~	~~85~~	~~86~~	~~87~~	~~88~~	89	~~90~~
~~91~~	~~92~~	~~93~~	~~94~~	~~95~~	~~96~~	97	~~98~~	~~99~~	~~100~~

https://github.com/danluu/setjmp-longjmp-ucontext-snippets/