

```

mytest:      file format elf64-littleaarch64

Disassembly of section .text:

00000000000073300 <main.addUpViaInterface>:
    }
}

//go:noinline
//go:nosplit
func addUpViaInterface(s []interface{}) {
    str    x30, [sp, #-48]!
    stur   x29, [sp, #-8]
    sub    x29, sp, #0x8
    str    x0, [sp, #56]
    for i := 0; i < len(s); i++ {
        mov    x2, xzr
    b     73338 <main.addUpViaInterface+0x38>
}

```

```

73338: cmp    x1, x2
7333c: b.le   73360 <main.addUpViaInterface+0x60>

```

```

sum += s[i].(int64)
73340: lsl    x3, x2, #4
73344: ldr    x3, [x0, x3]
73348: add    x4, x0, x2, lsl #4
7334c: adrp   x5, 85000 <type: *+0x5000>
73350: add    x5, x5, #0x300
73354: cmp    x3, x5
73358: b.eq   73318 <main.addUpViaInterface+0x18> // b.none

```

```

}
73360: ldp    x29, x30, [sp, #-8]
73364: add    sp, sp, #0x30
73368: ret
sum += s[i].(int64)

```

```

sum += s[i].(int64)
73318: adrp   x27, 12a000 <runtime.mheap_+0x18c20>
7331c: ldr    x3, [x27, #416]
73320: ldr    x4, [x4, #8]
73324: ldr    x4, [x4]
73328: add    x3, x3, x4
7332c: adrp   x27, 12a000 <runtime.mheap_+0x18c20>
73330: str    x3, [x27, #416]
for i := 0; i < len(s); i++ {
73334: add    x2, x2, #0x1

```

```

7335c: b     7336c <main.addUpViaInterface+0x6c>

```

```

7336c: mov    x0, x3
73370: mov    x1, x5
73374: adrp   x2, 86000 <type: *+0x6000>
73378: add    x2, x2, #0xca0
7337c: bl    17e70 <runtime.panicdottypeE>
73380: nop
...

```