argv[1][0]

argv[3][2]=='.'

./numint -m 0.0 10.0 5

argv[0] argv[1] argv[2] argv[3] argv[4]

| '1' | '0' | '.' | '0' | '\0' |
|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 |

char** argv

---

argv [0] [1]

char**

char*

char

| 600 | 675 | ..... |
|---|---|---|
| 695 | | |

argv is 695
argv[0] is 600
argv[0][1] is '/'

char* is an address of a char

| '.' | '/' | n | u | m | i | n | t | '\0' |
|---|---|---|---|---|---|---|---|---|
| 600 | 601 | | | | | | | |

| '-' | m | '\0' |
|---|---|---|
| 675 | | |

```
char *s = ........
// "If the string at s is the same string as "happy"..."
if(strcmp(s, "happy") == 0) {
```

↑ string compare

strcmp(a, b) returns 0 if and only if the strings at a and b are lexographically identical (i.e. "ab" and "ab") even if they are in different places in memory.

264

call stack (office hours)

9-7-2017

```
void bar(int a) {
    int n = 100;
    while (a!=0) {
        n = .....;
    }
    foo(n)
    int b = a;
}
```

```
int copy_of_a = a;
```

```
for(int string=0;
    string < structure;
    string ++) {
    char i[] = "Yo.";
}
```