

Pointers



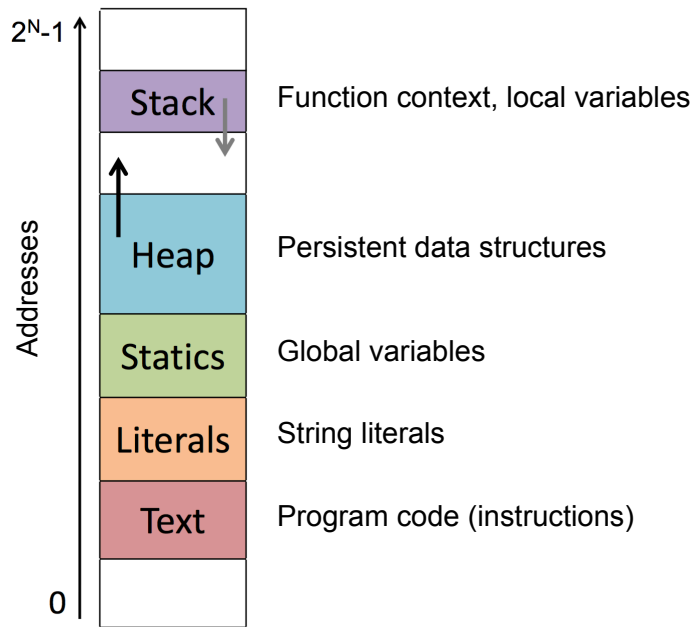
Null-Terminated Strings

null character



0x43	0x53	0x43	0x49	0x20	0x32	0x33	0x33	0x30	0x00
'C'	'S'	'C'	'I'	' '	'2'	'3'	'3'	'0'	'\0'

Memory Layout



Dynamic Memory Allocation Example

```
#define ZIP_LENGTH 5

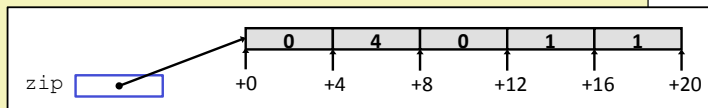
int* zip = (int*) malloc(sizeof(int) * ZIP_LENGTH);
if (zip == NULL) {
    perror("malloc failed");
    exit(0);
}

zip[0] = 0;
zip[1] = 4;
zip[2] = 0;
zip[3] = 1;
zip[4] = 1;

printf("zip is");
for (int i = 0; i < ZIP_LENGTH; i++) {
    printf(" %d", zip[i]);
}
printf("\n");

free(zip);
```

zip	0x7fedd240dc0	0x7ff58bdd938
	0	0x7fedd240dd0
	4	0x7fedd240dcc
	0	0x7fedd240dc8
	1	0x7fedd240dc4
	1	0x7fedd240dc0



Pointers to Pointers to ...

```
#define NUM_ZIPS 3

int** zips = (int**) malloc(sizeof(int*) * NUM_ZIPS);
...
zips[0] = (int*) malloc(sizeof(int) * ZIP_LENGTH);
...
int* first_zip = zips[0];
first_zip[0] = 0;
zips[0][1] = 4;
zips[0][2] = 0;
first_zip[3] = 1;
zips[0][4] = 1;
```

