CS 208

Wed, 11 Jan 2023
program instructions (read-only)

"moose"

global initialized data (read-only)

"heap" memory allocated by malloc

local variables
Variables

int j

1. "Declaration" — there is a variable named j of type int

2. "Definition" — hey run-time system, set aside 4 bytes to hold j (sizeof(int))
int j = 5;

Declare (int, j)

Definition (4 bytes allocated)

Initialization (puts 5 into those 4 bytes)
& operator (unary operator)
"the address of"

int j = 5;
int *p = &j;

2 "pointer to" an int
"address of" an int

*p = 47;

 Dereferencing p
char *p;
p = (char *) malloc(100 * sizeof(char));
if (p == NULL) {
    uh-oh
    exit(1);
}
strncpy(p, "coati mundi", 100);
// Done w/ p?
free(p);
typedef unsigned int size_t

size_t j;
unsigned int j;

identical
struct Circle {
    int center_x;
    int center_y;
    int radius;
};

struct Circle c;
    c.center_x = 5;
    c.radius = 23;
typedef struct circle
    int center_x;
    int center_y;
    int radius;

#define circle C
C.center_x = 5;
circle \( c_j \)
circle \(*p = 8c_j\) 
\((*p)\). center-_x = 5;
p \rightarrow center-_y = 19.