C: structs; Number representation: binary

COSC 208, Introduction to Computer Systems, 2021-09-08

Warm-up

Q1: *Write a function called* `count_words` *that takes a string and counts the number of words in the string. Assume each word is separated by a single space, and the string will contain at least one word. For example,* `"Today is Wednesday."` *contains 3 words.*

🔴 Stop here after completing the warm-up; if you have extra time please **skip ahead** to the extra practice.

Structs

Q2: *What is the output of this program?*

```
struct one {
    char x;
    char y;
    short z;
};
struct two {
    int m;
    int n[10];
};
int main() {
    struct one a;
    struct two b;
    printf("%d %d\n", sizeof(struct one), sizeof(a.z));
    printf("%d %d\n", sizeof(b), sizeof(b.n));
}
```

Q3: *What is the output of this program?*

```c
struct alpha {
    char x[10];
    int y;
};
struct beta {
    int b;
    int c;
};
int main() {
    struct alpha a = { "Colgate", 13 };
    struct beta b = { 1, 2 };
    struct beta c = { 3, 4 };
    a.y += -13;
    b.b = 5;
    c = b;
    b.c = 6;
    printf("a %s %d\n", a.x, a.y);
    printf("b %d %d\n", b.b, b.c);
    printf("c %d %d\n", c.b, c.c);
}
```

- Q4: *Draw the stack right before the return from* `mystery`

```c
struct personT {
  char name[32];
  int  age;
};

void mystery(int i_val, struct personT per, int a[], int n);

int main() {
  struct personT person;
  int x, i;
  int arr[5];

  for(i=0; i < 5; i++) {
     printf("arr[%d] = %d\n", i, arr[i]);
  }
  x = 13;
  strcpy(person.name, "Lila");
  person.age = 10;
  mystery(x, person, arr, 5);

  for(i=0; i < 5; i++) {
     printf("arr[%d] = %d\n", i, arr[i], 5);
  }
  printf("x = %d age = %d name = %s\n", x, person.age, person.name);

}

void mystery(int i_val, struct personT per, int a[], int n) {
  for(int i = 0; i < n; i++) {
     a[i] = a[i]*a[i];
  }
  strcpy(per.name, "Orso");
  per.age = 18;
  i_val = 100;
  //**** DRAW STACK IS RIGHT BEFORE return STATEMENT IS EXECUTED
  return;
}
```

Binary (i.e., base 2)

Convert these binary numbers to decimal (i.e., base 10):

Q5: `0b10`

Q6: `0b11`

Q7: `0b1010`

Q8: `0b1111`

Q9: `0b11001100`

Extra practice

Q10: *Write a struct definition to represent a date (year, month number, and day).*

Q11: *Write a function called* `compare` *that takes two date structs and returns -1 if the first date occurs before the second, 0 if the dates are equal, and 1 if the first date occurs after the second.*