

Question 1. [3 points] What output is printed by the following program (which begins on the left and continues on the right)?

| | |
|---|---|
| <pre>#include <stdio.h> void halveIt(int x) { x = x / 2; }</pre> | <pre>int main(void) { int y = 8; halveIt(y); printf("%i\n", y); return 0; }</pre> |
|---|---|

A. 4

B. 8

C. The output can't be predicted

D. The program does not compile

Question 2. [3 points] What output is printed by the following program (which begins on the left and continues on the right)?

| | |
|---|--|
| <pre>#include <stdio.h> void halveThem(int a[], int n) { for (int i = 0; i < n; i++) { a[i] = a[i] / 2; } }</pre> | <pre>int main(void) { int nums[4] = { 2, 4, 6, 8 }; halveThem(nums, 4); for (int i = 0; i < 4; i++) { printf("%i ", nums[i]); } return 0; }</pre> |
|---|--|

A. 1 2 3 4

B. 2 4 6 8

C. The output can't be predicted

D. The program does not compile

Question 3. [3 points] What output is printed by the following program (which begins on the left and continues on the right)?

| | |
|---|---|
| <pre>#include <stdio.h> struct Box { int side; }; struct Box halveIt(struct Box b) { b = b / 2; return b; }</pre> | <pre>int main(void) { struct Box mine; mine = 42; mine = halveIt(mine); printf("%i\n", mine.x); return 0; }</pre> |
|---|---|

- A. 21
- B. 42
- C. The output can't be predicted
- D. The program does not compile

Question 4. [3 points] What output is printed by the following program (which begins on the left and continues on the right)?

| | |
|--|---|
| <pre>#include <stdio.h> struct Stuff { int a; float b; }; struct Stuff doubleStuff(float x, int y) { struct Stuff s; s.a = y; s.b = 2*x; return s; }</pre> | <pre>int main(void) { struct Stuff oreo; int cookie = 2; float filling = 4; oreo = doubleStuff(filling, cookie); printf("%i %.1f\n", oreo.a, oreo.b); return 0; }</pre> |
|--|---|

Handwritten annotations: A blue arrow points from the `doubleStuff` function call in `main` to the function definition. Another blue arrow points from the `printf` statement to the `oreo` variable. A handwritten box labeled `oreo` contains the values `a 2` and `b 8.0`. The number `4.0` is written above the `filling` variable, and the number `2` is written below the `cookie` variable.

- A. 2 4.0
- B. 4 4.0
- C. 2 8.0
- D. The program does not compile

Question 5. [6 points] The following program reads a single integer value (n). Complete the program so that the `printf` statement in `main` prints the value 2^n (2 raised to the power n). You can assume n will be non-negative. You must:

1. Add a call to `multiplyByTwo` to the for loop in `main`
2. Complete the definition of the `multiplyByTwo` function

```
#include <stdio.h>

void multiplyByTwo(int *x);

int main(void) {
    int n;
    scanf("%i", &n);

    int product = 1;
    for (int i = 1; i <= n; i++) {
        multiplyByTwo(&product);
    }

    printf("%i\n", product);
    return 0;
}

void multiplyByTwo(int *x) {
    *x = *x * 2;
}

```

Question 6. [3 points] Create a struct type called `Student` that has member fields to store the a student's age, GPA, sex (either 'M' or 'F'), and number of credits earned. Use appropriate data types and meaningful variable names for each.

```
struct Student {  
    int age;  
    double gpa;  
    char sex;  
    int num_credits;  
};
```

Question 7. [3 points] Declare a variable of type `Student` (defined in Question 6) and assign values of your choice to the member fields.

```
struct Student example;  
example.age = 21;  
example.gpa = 3.91;  
example.sex = 'F';  
example.num_credits = 94;
```

Question 8. [4 points] Define a function called `print_student_info()` that takes a `struct Student` as a parameter and prints the student's age, GPA, sex, and number of credits earned in the following format:

age, GPA, sex, numberOfCredits

For example, for a 20-year old student, the output might be

20,3.71,F,45

```
void print_student_info(struct Student s) {  
    printf("%i, %.2lf, %c, %i",  
        s.age, s.gpa, s.sex, s.num_credits);  
}
```

For Questions 9–14, circle True or False.

Question 9. [2 points] True or False: It is possible to return an array from a function as a return value.

Question 10. [2 points] True or False: If `a` is an array parameter, it is possible to find out how many elements `a` has using the syntax `a.length`.

Question 11. [2 points] True or False: Structs allow you to define new data types.

Question 12. [2 points] True or False: It is permissible to assign the value of one struct variable to another struct variable, as long as the variables have the same types.

Question 13. [2 points] True or False: The ampersand (`&`) is the “address of” operator.

Question 14. [2 points] True or False: The asterisk (`*`) is the “address of” operator.