

Question 1. [2 points] Circle **True** or **False**: A function must have at least 1 parameter.

Question 2. [2 points] Circle **True** or **False**: Every **if** statement must have an **else** block.

Question 3. [2 points] Circle **True** or **False**: Every **else** block must have a corresponding **if** block.

Question 4. [2 points] Fill in the blank: the _____ operator, when applied to a variable, yields the address of the variable.

Question 5. [2 points] Fill in the blank: the _____ operator, when applied to a pointer, yields the variable the pointer points to.

Question 6. [8 points] Describe the 4 errors in the following code, which is meant to compute the sum of the integers from 0 to a maximum integer value:

```
int sum, imax, i

printf("Enter the maximum integer to sum: ");
scanf("%i", &imax);

for (i = 0; i <= imax; ) {
    sum += i;
}

printf("Their sum is %i.\n", sum);
```

Question 7. [6 points] Write the prototype and definition of a function that takes two integer values x and y as parameters, computes $5x^2 - 3y$, and returns the result.

Prototype:

Definition:

Question 8. [3 points] What output is printed by the following code?

```
int a, b;

a = 4;
b = a;
printf("%i\n", b);

a = 5;
printf("%i\n", b);
```

Question 9. [3 points] What output is printed by the following code?

```
int a = 7;
int b = 3;

printf("%i %i\n", a / b, a % b);
```

Question 10. [3 points] What output is printed by the following program (which begins on the left and continues on the right)?

<pre>#include <stdio.h> int f(int x) { x = x + 1; return x; }</pre>	<pre>int main(void) { int y = 3; f(y); printf("%i\n", y); return 0; }</pre>
--	---

Question 11. [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 f(int *p) { *p = *p + 1; }</pre>	<pre>int main(void) { int y = 3; f(&y); printf("%i\n", y); return 0; }</pre>
--	--

Question 12. [3 points] What output is printed by the following program (which begins on the left and continues on the right)?

Hint: If p and q are pointers, $p = q$ means “make p point to the variable that q points to”.

<pre>#include <stdio.h> void f(int *p, int *q) { int *tmp = p; p = q; q = tmp; }</pre>	<pre>int main(void) { int a = 3, b = 4; f(&a, &b); printf("%i %i\n", a, b); return 0; }</pre>
---	---

Question 13. [5 points] Show the definition for a data type called `struct BankAccount` with the following fields:

- `id`, an integer
- `type`, which should allow the values 'S' for savings and 'C' for checking
- `balance`, which represents an amount of money in whole pennies (*e.g.*, if the account balance is \$404.31, this field should contain the value 40431)

Question 14. [6 points] Consider the following struct type definition:

```
struct Point {
    double x, y;
};
```

Show the prototype and definition of a function called `transpose` that takes a `struct Point` as a parameter and returns a `struct Point` that exchanges the `x` and `y` values of the parameter.

Example:

```
struct Point p, q;
p.x = 3.33;
p.y = 4.44;

q = transpose(p);
printf("x=%.2lf, y=%.2lf\n", q.x, q.y); // prints x=4.44, y=3.33
```

Prototype:

Definition:

Programming Questions

Note: For all of the programming questions, you should use `scanf` to read the input value(s) required by the program.

Note: Make sure your programs produce the output in **exactly** the format described, including capitalization and punctuation. You may not receive credit for programs that produce incorrectly-formatted output.

Getting started: Start **Cygwin Terminal** and **Notepad++**. (Note: do *not* open any other programs.) Your instructor will give you the name of a zip file. In Cygwin Terminal, run the following commands:

```
cd h:
mkdir -p CS101_Final
cd CS101
curl -O http://faculty.ycp.edu/~dhovemey/spring2014/cs101/assign/zipfile
unzip zipfile
cd CS101_Final
```

Note that in the `curl` command, the `-O` has the letter ‘O’, not the digit ‘0’.

Substitute the name of the zip file for *zipfile*.

Editing code: Use Notepad++ to open the source file (e.g., `question15.cpp`) referred to in the question. Do not open any files other than the ones for the exam.

Compiling: To compile the program for Question 15, run the following command in Cygwin Terminal:

```
make question15.exe
```

Change the number as appropriate for the other questions (e.g., `question16.exe`).

Running: To run the program for Question 15, run the following command in Cygwin Terminal:

```
./question15.exe
```

Change the number as appropriate for the other questions (e.g., `question16.exe`).

To submit: In Cygwin Terminal, run the command

```
make submit
```

Enter your Marmoset username and password when prompted.

Good luck!

Question 15. [15 points] The program in `question15.cpp` reads a `double` value `x` and an `int` value `n`, and prints a `double` value which is the result of raising `x` to the power `n`.

Your task is to add a prototype and definition for a function called `Power`, which computes the result of raising `x` to the power `n`.

Important: Do not modify the `main` function in any way.

```
result = Power(x, n);
```

Note that if `n` is a non-negative integer the function returns the value x^n and if `n` is a negative integer the function returns the value $1/x^{|n|}$ (1 divided by x raised to the absolute value of n).

Example runs (user input in **bold**):

Enter x: 3.0 Enter n: 2 Result is 9.000000	Enter x: 3.0 Enter n: -2 Result is 0.111111
--	---

Important: do *not* use the built-in `pow` function.

Hint: Raising a number to an integer power involves repeated multiplication. So, to raise x to the power n , use a loop that executes $|n|$ times. If n is positive, the factors being multiplied are equal to x . If n is negative, the factors being multiplied are equal to $1/x$.

Hint: When computing a product of factors, your product variable should be initialized to 1.

Question 16. [10 points] Complete the program in `question16.cpp` so that it prints a figure using `X` and `-` characters based on the integer input value as shown in the examples below (user input in **bold**):

Size: 5 X---X -X-X- --X-- -X-X- X---X	Size: 7 X-----X -X---X- --X-X-- ---X--- --X-X-- -X---X- X-----X
---	---

You can assume that the input value will be positive and odd.

Question 17. [15 points] The program in `question17.cpp` reads a sequence of non-negative integer values (using `scanf`) and stores them in an array. The first negative value read indicates the end of the sequence. It then calls a function called `find_min` to find the minimum of all of the values in the array.

Your task is to add a prototype and definition for the `find_min` function. It takes two parameters: an array of integers, and an integer which specifies how many elements are stored in the array. It should return the minimum value of the elements in the array.

After you add the prototype and definition for `find_min`, the program's output should be a line of the form:

```
Minimum is: N
```

where N is the minimum of the nonnegative integers.

For example, if the program reads the input

```
74 7 45 97 50 -1
```

Then it should print the output

```
Minimum is: 7
```

Try entering the input as shown above. Verify that your program's result is identical.

Important: Do not modify the `main` function in any way.

Question 18. [10 points] Consider the following struct type:

```
struct Card {
    int rank;
    char suit;
};
```

An instance of `struct Card` represents a playing card, where `rank` represents the rank (2–10 for numbered cards, 11 for Jack, 12 for Queen, 13 for King, and 14 for Ace) and `suit` represents the suit ('C', 'D', 'H', 'S' for Clubs, Diamonds, Hearts, and Spades).

Complete the program in `question18.cpp` by completing the definitions of the `is_flush` and `is_straight` functions. Both functions take two parameters: `hand`, an array of `struct Card` elements, and `numcards`, an integer indicating how many cards there are in the hand. Both functions return a `bool` (true or false) value.

`is_flush` should return `true` if all cards in the array have the same suit, `false` otherwise.

`is_straight` should return `true` if the cards in the array are arranged in a sequence such that each subsequent card has a rank that is exactly one greater than its predecessor. (Note that you may assume that the cards will be arranged in increasing order by rank.)

The `main` function allows the user to enter a sequence of cards by rank and suit, with the special rank value -1 indicating that there are no more cards to add. It then calls the `is_flush` and `is_straight` functions and prints the results.

Example runs (user input in **bold**):

Enter cards: 3 C 4 D 5 C 6 H 7 S -1 is_flush returned false is_straight returned true	Enter cards: 2 H 3 H 8 H 11 H -1 is_flush returned true is_straight returned false
--	---

Important: Do not modify the main function in any way.