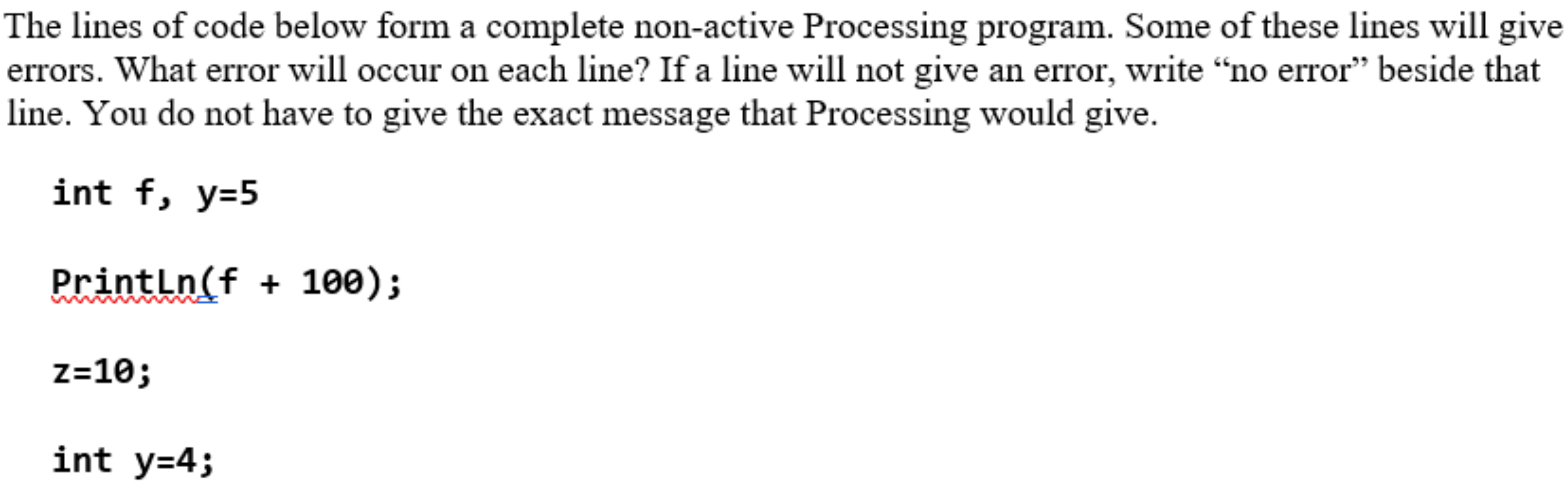
SOME EXTRA SHORT ANSWER QUESTIONS TO TRY AT HOME

SHORT ANSWER: BASICS



The lines of code below form a complete non-active Processing program. Some of these lines will give errors. What error will occur on each line? If a line will not give an error, write “no error” beside that line. You do not have to give the exact message that Processing would give.

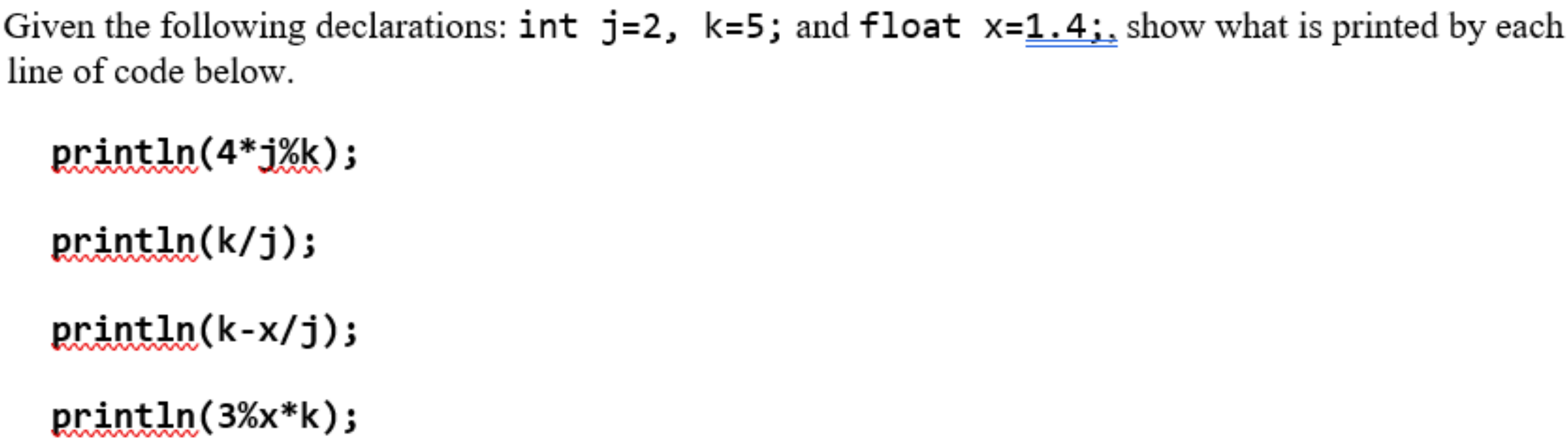
```
int f, y=5

PrintLn(f + 100);

z=10;

int y=4;
```

SHORT ANSWER: ACTIVE/TYPES/SCOPE



Given the following declarations: `int j=2, k=5;` and `float x=1.4;`, show what is printed by each line of code below.

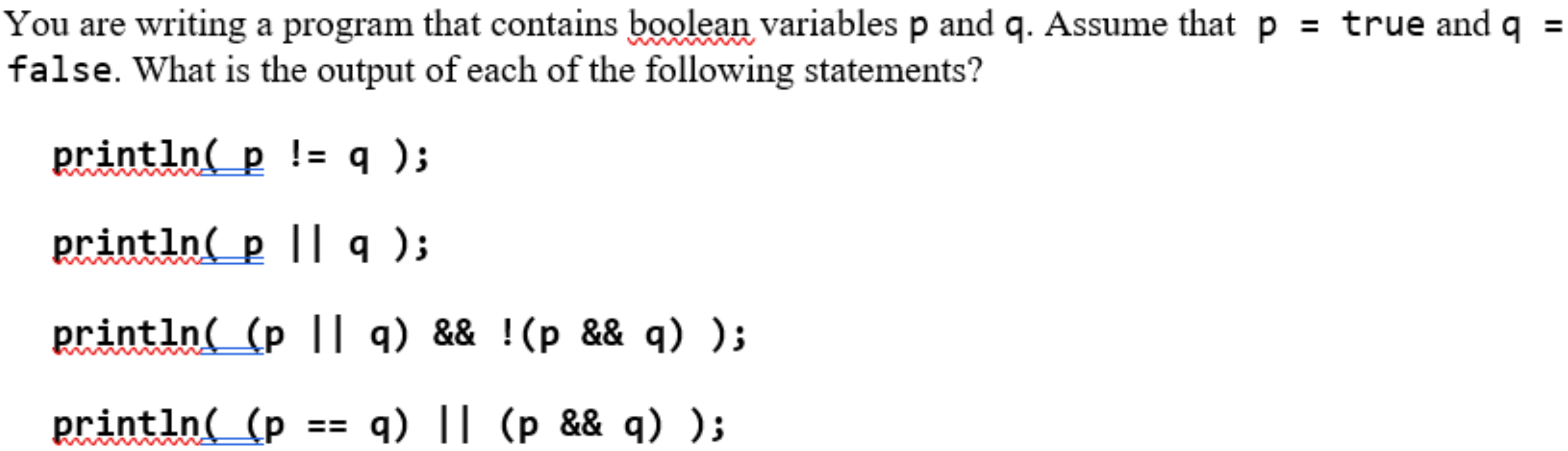
```
println(4*j%k);

println(k/j);

println(k-x/j);

println(3%x*k);
```

SHORT ANSWER: BOOLEAN/IF/CONDITIONALS



You are writing a program that contains boolean variables `p` and `q`. Assume that `p = true` and `q = false`. What is the output of each of the following statements?

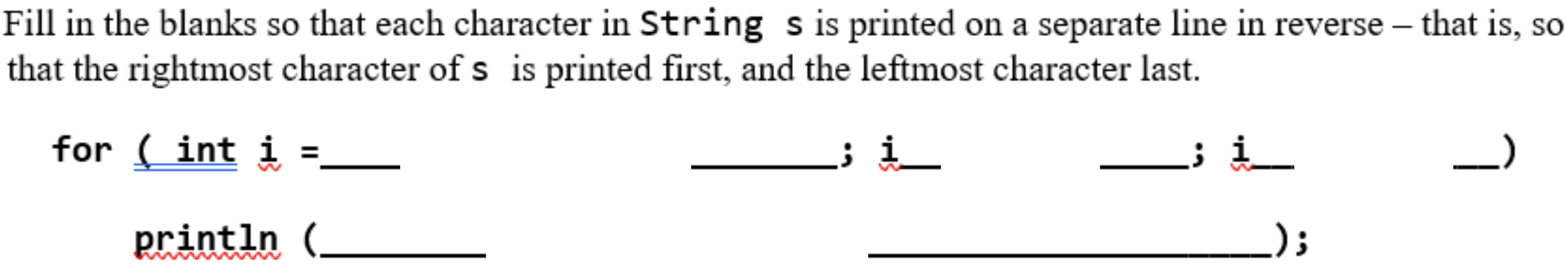
```
println( p != q );

println( p || q );

println( (p || q) && !(p && q) );

println( (p == q) || (p && q) );
```

SHORT ANSWER: STRINGS/CHARS

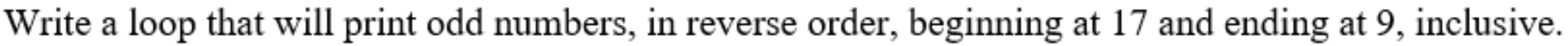


Fill in the blanks so that each character in `String s` is printed on a separate line in reverse – that is, so that the rightmost character of `s` is printed first, and the leftmost character last.

```
for ( int i =____                    _________; i___          _____; i___          ___)

     println (__________                        _________________________);
```

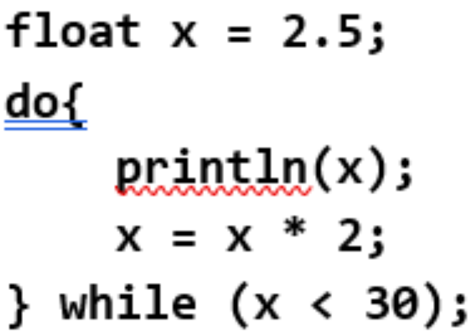
SHORT ANSWER: LOOPS



Write a loop that will print odd numbers, in reverse order, beginning at 17 and ending at 9, inclusive.

SHORT ANSWER: LOOPS

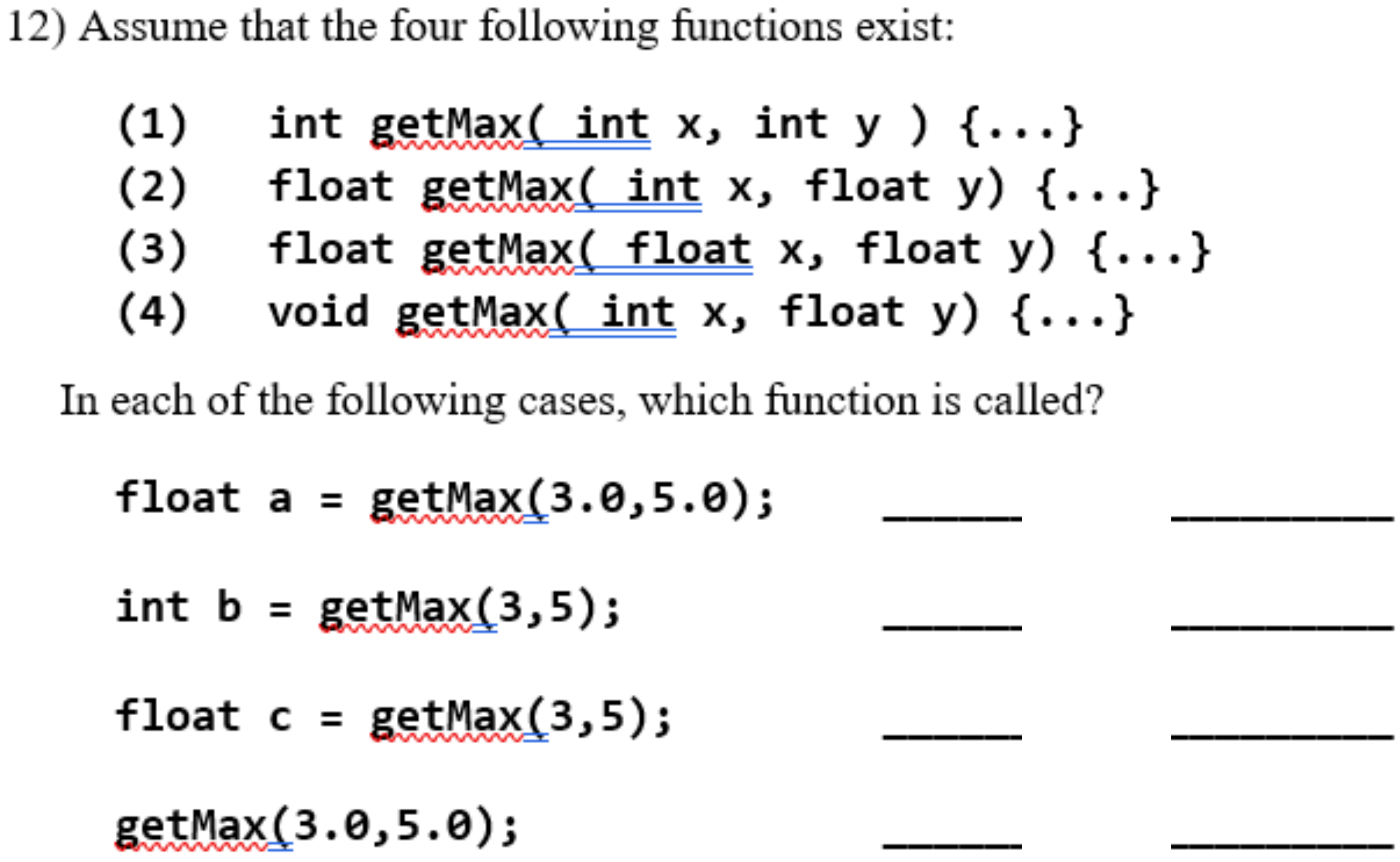
Show the output of the following code.



```
float x = 2.5;
do{
    println(x);
    x = x * 2;
} while (x < 30);
```

Output:

SHORT ANSWER: FUNCTIONS



12) Assume that the four following functions exist:

```
(1)   int getMax( int x, int y ) {...}
(2)   float getMax( int x, float y) {...}
(3)   float getMax( float x, float y) {...}
(4)   void getMax( int x, float y) {...}
```

In each of the following cases, which function is called?

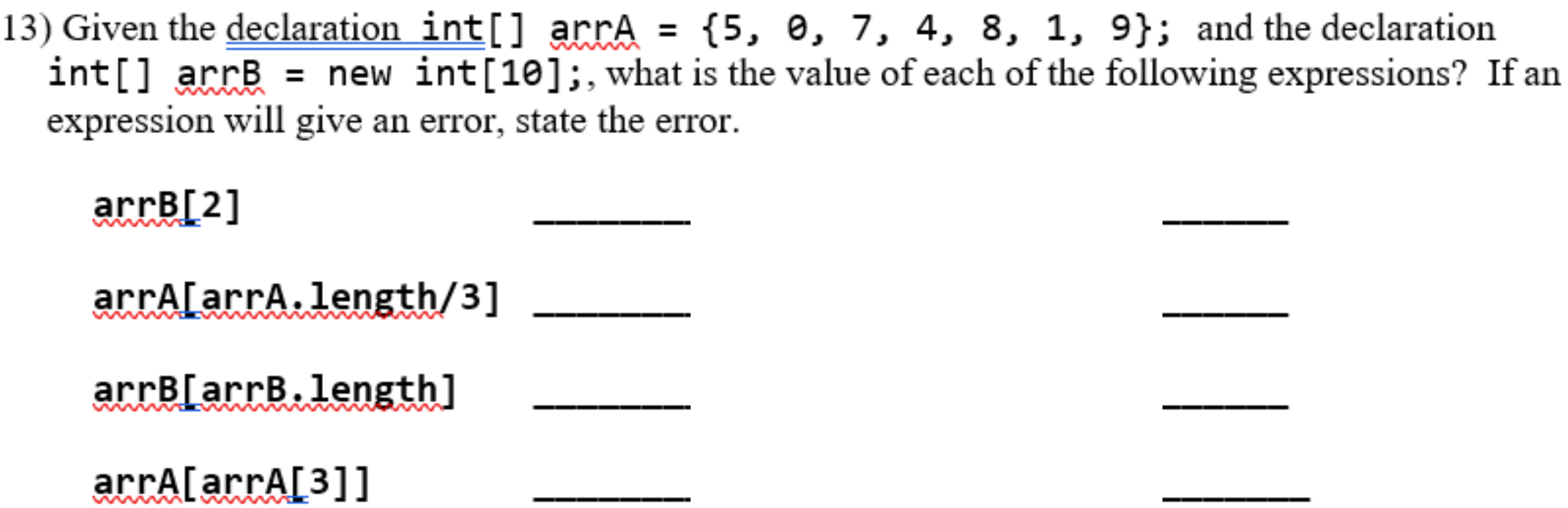
```
float a = getMax(3.0,5.0);        ______     _________

int b = getMax(3,5);              ______     _________

float c = getMax(3,5);            ______     _________

getMax(3.0,5.0);                  ______     _________
```

SHORT ANSWER: ARRAYS



13) Given the declaration `int[] arrA = {5, 0, 7, 4, 8, 1, 9};` and the declaration `int[] arrB = new int[10];`, what is the value of each of the following expressions? If an expression will give an error, state the error.

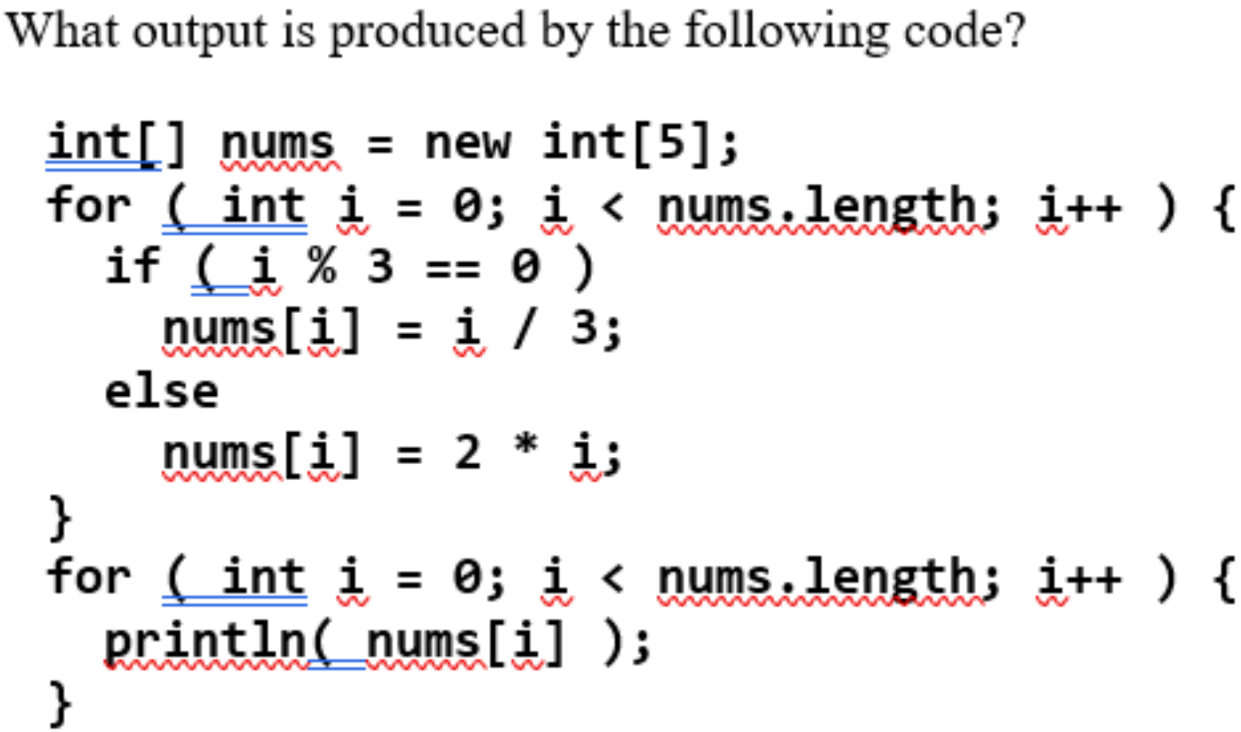
`arrB[2]` ________ ________

`arrA[arrA.length/3]` ________ ________

`arrB[arrB.length]` ________ ________

`arrA[arrA[3]]` ________ ________

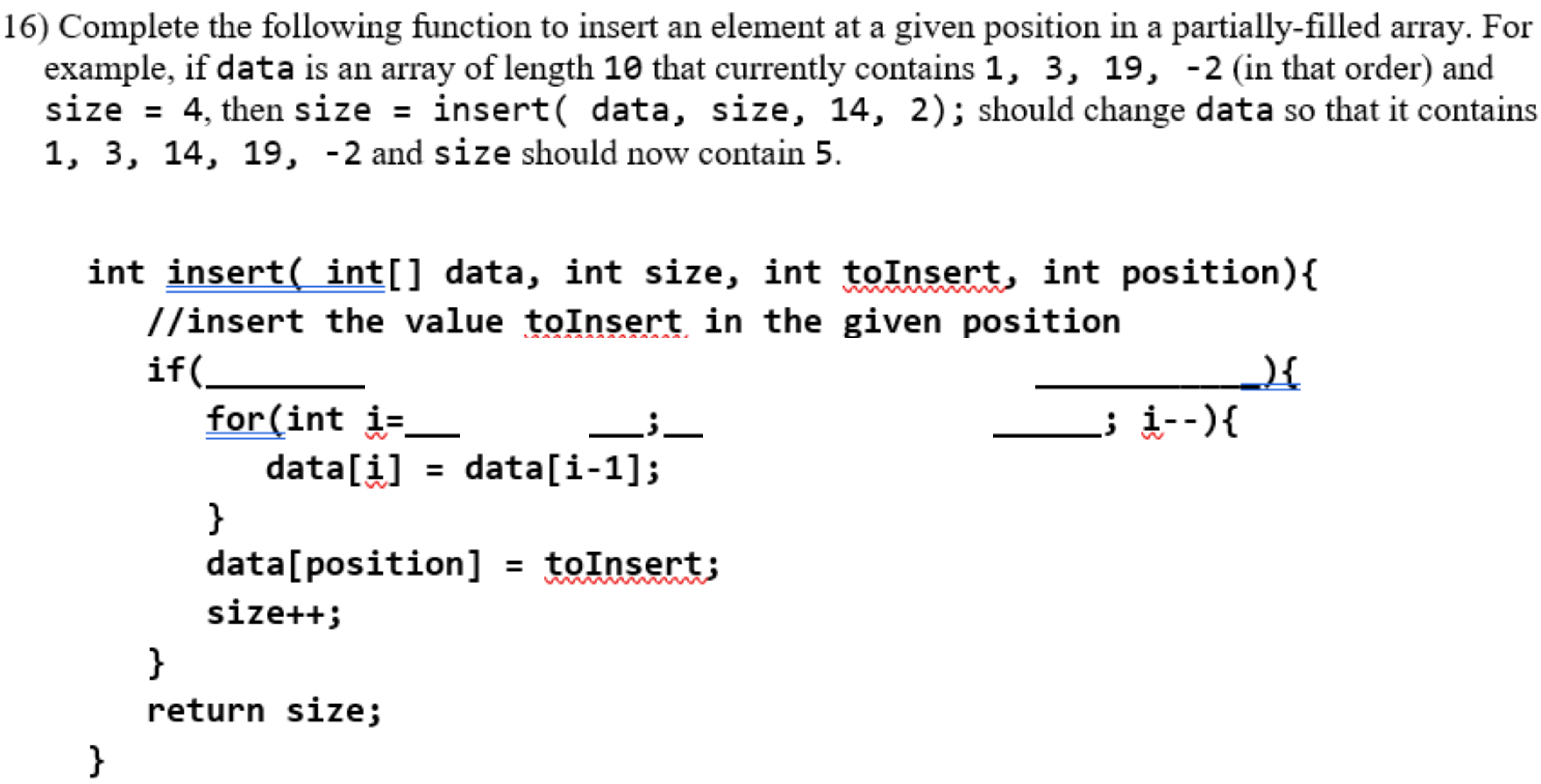
SHORT ANSWER: ARRAYS



What output is produced by the following code?

```
int[] nums = new int[5];
for ( int i = 0; i < nums.length; i++ ) {
  if ( i % 3 == 0 )
    nums[i] = i / 3;
  else
    nums[i] = 2 * i;
}
for ( int i = 0; i < nums.length; i++ ) {
  println( nums[i] );
}
```

SHORT ANSWER: PARTIALLY-FULL ARRAYS



16) Complete the following function to insert an element at a given position in a partially-filled array. For example, if `data` is an array of length `10` that currently contains `1, 3, 19, -2` (in that order) and `size = 4`, then `size = insert( data, size, 14, 2);` should change `data` so that it contains `1, 3, 14, 19, -2` and `size` should now contain `5`.

```
int insert( int[] data, int size, int toInsert, int position){
   //insert the value toInsert in the given position
   if(________                                   ___________){
      for(int i=___      ___;__                   _____; i--){
         data[i] = data[i-1];
      }
      data[position] = toInsert;
      size++;
   }
   return size;
}
```

SHORT ANSWER: BASICS

Draw the image that will appear in the canvas after running this program.

```
size(400,400);
background(255);
line(0,0,400,400);
ellipse(200,200,200,200);
rect(200,200,100,100);
```

SHORT ANSWER: ACTIVE/TYPES/SCOPE/ETC.

What output is printed in the console by the following program?

```
int a = 2;
int b = 11;
float c = float(b/a);
println(c);
c += 5.0/10;
println(int(c));
```

SHORT ANSWER: BOOLEAN & IF

Fill in the blank to complete the ***single*** line of code that will match the following comment. Assume that **a** and **b** have already been declared and assigned values.

```
//Set c to true if either a or b are true, but not both.
```

boolean c = ____;

SHORT ANSWER: STRINGS & CHARS

Fill in the blanks in the code below so that it will count the number of times that the character **`'X'`** appears in a **`String s`**. Assume that **`s`** was defined and given a value earlier.

```
int count = 0;
for(_________________________________)
   if(________________________________)
      count++;
```

SHORT ANSWER: LOOPS

What output is printed in the console by the following program?

```
for(int i = 0; i < 5; i++)
   print("+");
println();
int i = 63;
while(i > 0){
   print("+");
   i /= 2;
}
println();
```

SHORT ANSWER: FUNCTIONS

What output is printed in the console by the following program? If there is an error in the program, show the error message, and circle the line that generates the error.

```
int y = 3, n = 2;
void setup(){
  int y = 5;
  int z = f1(7);
  println(z);
  println(f2(y+3));
}
int f1(int n){
  return n%2;
}
int f2(int y){
  return y*3;
}
```

SHORT ANSWER: ARRAYS

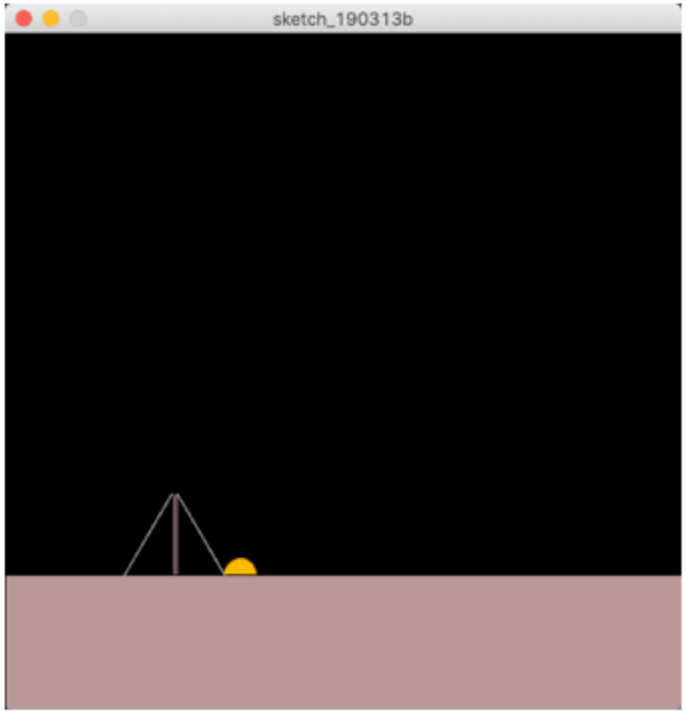
Write the code to create an array named **a** with length **N** and containing the values **1** to **N**.

SOME EXTRA SAMPLE PROGRAMMING QUESTIONS

17) Write a non-active `Processing` program that will draw the image shown at right. This is a simple desert camping scene with a dark starless night, a tent, and a fire to keep warm.

- The desert ground is one-fifth the height of the canvas.
- The tent is centered one-quarter of the way across the canvas.
- The tent is 75 pixels wide, and 60 pixels tall. Draw the tent as a rectangle (the pole in the center is 5 pixels wide) and two lines.
- The fire is a circle of diameter 25.

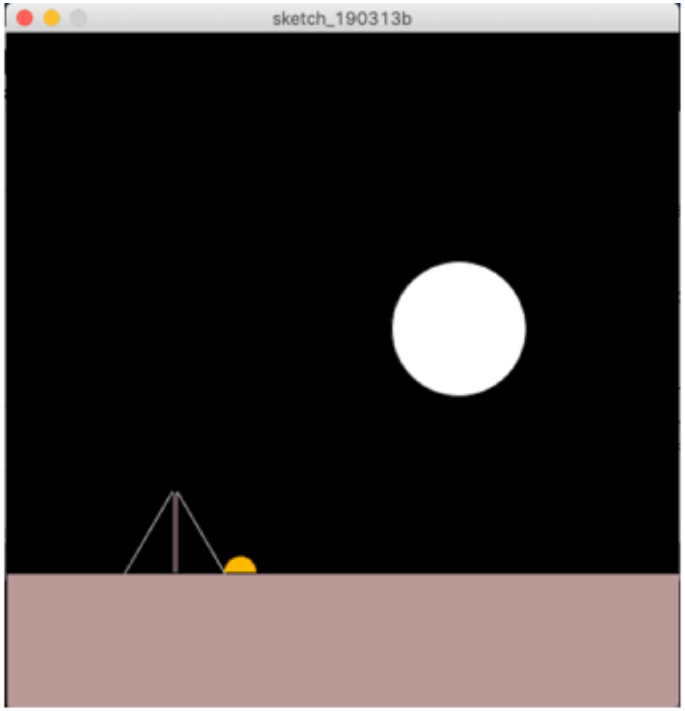
You must make use of the named constants below to draw the image. Draw everything required using only the variables supplied. Do not use any magic numbers.



```
size(500,500);

final int GROUND_HEIGHT = _______          _______;
final int TENT_X = ________             ___;
final int POLE_WIDTH = ___             _;
final int TENT_WIDTH = ___             __;
final int TENT_HEIGHT = __             ___;
final int FIRE_DIAMETER =              _____;
final int GROUND_COLOUR = #B49999, TENT_POLE_COLOUR = #675555,
          TENT_COLOUR = #E8E8E8, FIRE_COLOUR = #FFBB27;
```

18) Add to the previous question to make the moon orbit slowly over the desert. To do this, create an active `Processing` program. Assume that all of the code, including constants, from the answer to the previous question has been moved into a function called `drawDesert()`, which you should call in `draw()` to draw the scene.



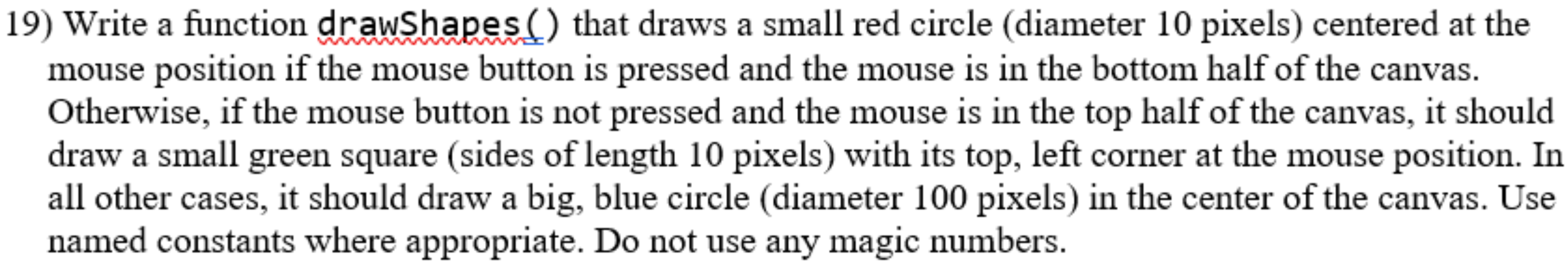
- The moon is 100 pixels wide.
- The moon is orbiting 200 pixels from the center of the desert horizon.
- The moon has a colour of 255 (white).

Make sure that you are drawing things in the correct order (the moon should be hidden by the desert when it is below the ground). Declare constants or variables, as appropriate, to store any dimensions. Do not use any magic numbers.

```
float angle = 0;
final int ORBIT_DISTANCE = _          ____;
final int MOON_DIAMETER = __          ___;
final int MOON_COLOUR = ____          _;
final float MOON_SPEED = TWO_PI/240;

void setup() {
  size(500,500);
}
void draw() {
  final int GROUND_HEIGHT = ______          _____;
```

PROGRAMMING: BOOLEAN & IF

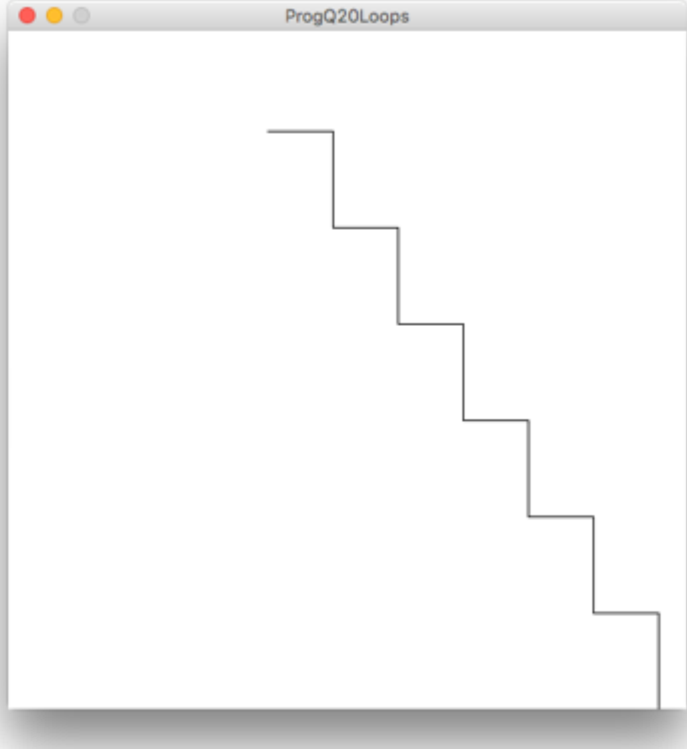


19) Write a function drawShapes() that draws a small red circle (diameter 10 pixels) centered at the mouse position if the mouse button is pressed and the mouse is in the bottom half of the canvas. Otherwise, if the mouse button is not pressed and the mouse is in the top half of the canvas, it should draw a small green square (sides of length 10 pixels) with its top, left corner at the mouse position. In all other cases, it should draw a big, blue circle (diameter 100 pixels) in the center of the canvas. Use named constants where appropriate. Do not use any magic numbers.

```
final int SMALL_DIAMETER = 10;
final int LARGE_DIAMETER = 100;
final int SQUARE_LENGTH = 10;
final int RED = #FF0000, GREEN = #00FF00, BLUE = #0000FF;
```

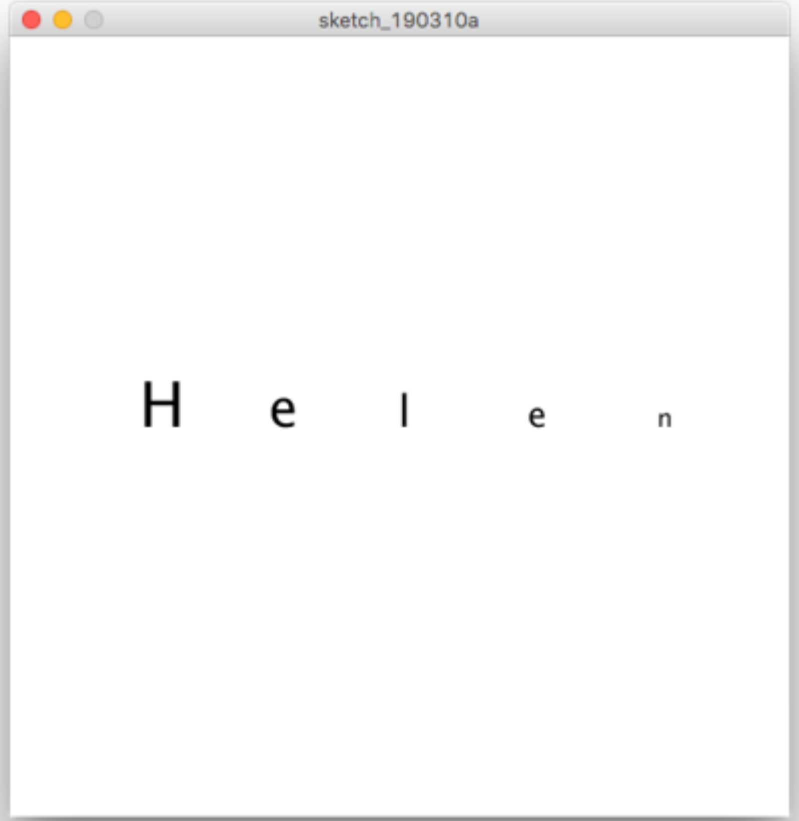
PROGRAMMING: LOOPS

20) Write a function drawStairs, that is passed a point, (x, y) and a number n. For example, a function call would be drawStairs(x, y, n);. The function should draw n steps starting at the given point, going down and to the right, and ending at the bottom of the canvas. Each step should be roughly the same height and width – at most one pixel different in height or width. See the picture at right for an example of what the function draws. This function should use only a single loop. **Hint**: The height of one step is the height of the staircase, divided by the number of steps. However, if you draw all the steps at this size, you will have pixels left over. How many pixels are left? How can you distribute those so that there is at most one pixel difference in the height of the steps?



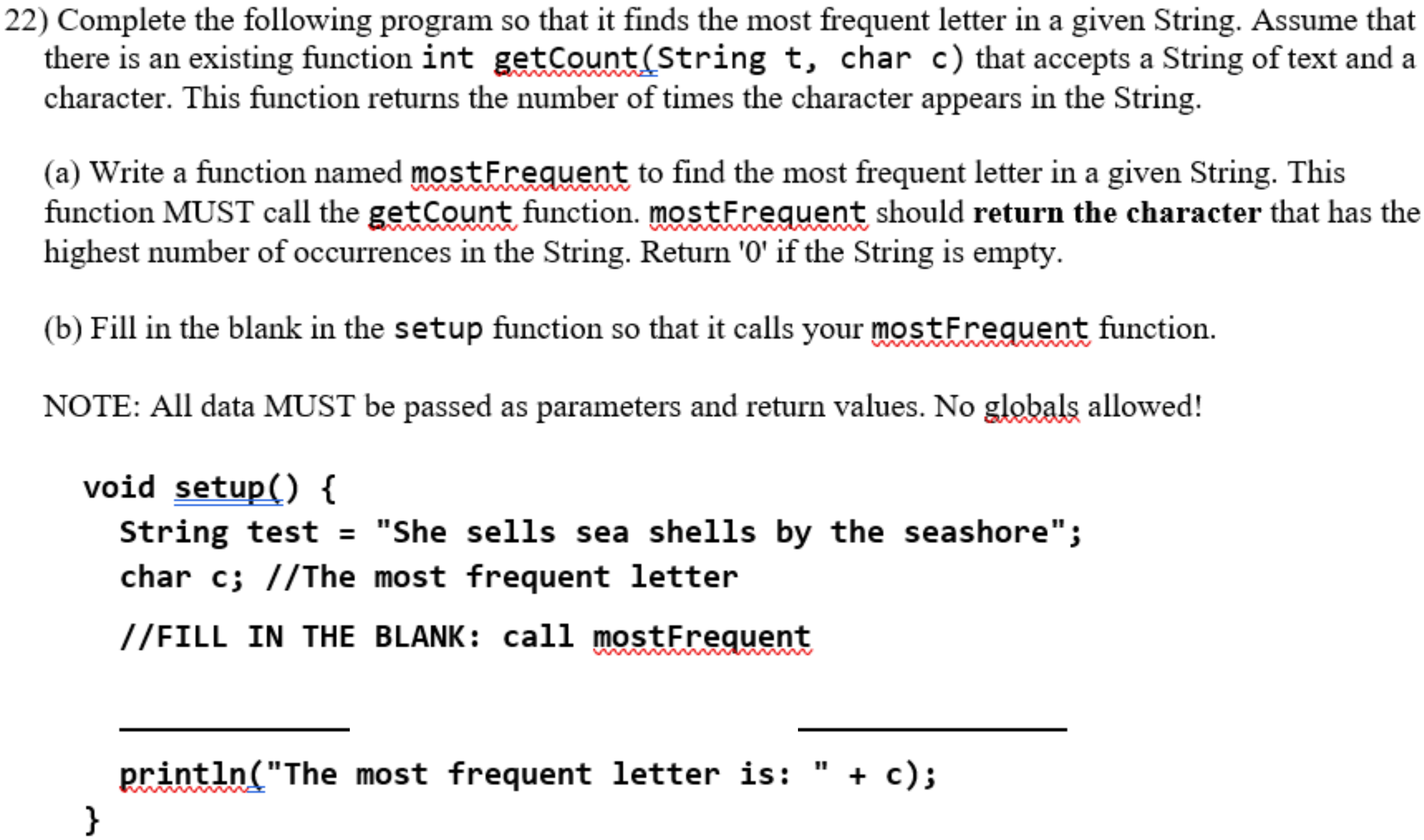
PROGRAMMING: STRINGS & CHARS

21) Write a function, `printShrinking`, that is passed a `String`. It should print the letters of the `String` individually, equally spaced on the center line. The size of the largest letter (the first one) should be 40 points, the size of the smallest should be 10 points, and the text size should decrease by equal amounts from letter to letter. The picture shows the results of passing the function "Helen".



```
H    e    l    e    n
```

PROGRAMMING: FUNCTIONS



22) Complete the following program so that it finds the most frequent letter in a given String. Assume that there is an existing function `int getCount(String t, char c)` that accepts a String of text and a character. This function returns the number of times the character appears in the String.

(a) Write a function named `mostFrequent` to find the most frequent letter in a given String. This function MUST call the `getCount` function. `mostFrequent` should **return the character** that has the highest number of occurrences in the String. Return '0' if the String is empty.

(b) Fill in the blank in the `setup` function so that it calls your `mostFrequent` function.

NOTE: All data MUST be passed as parameters and return values. No globals allowed!

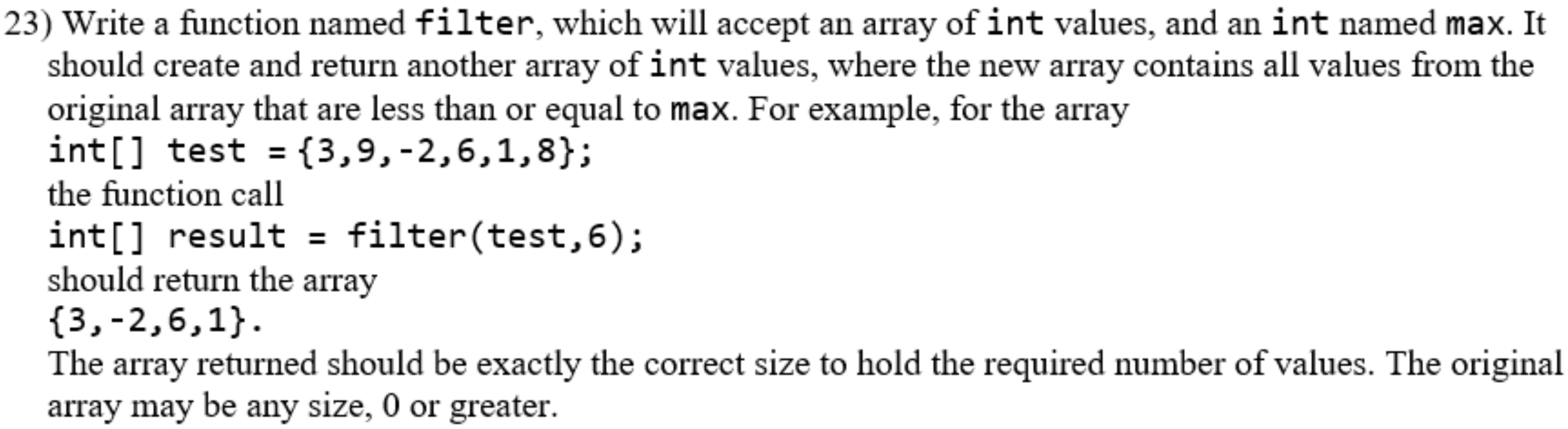
```
void setup() {
  String test = "She sells sea shells by the seashore";
  char c; //The most frequent letter

  //FILL IN THE BLANK: call mostFrequent

  ____________                    ______________

  println("The most frequent letter is: " + c);
}
```

PROGRAMMING: ARRAYS



23) Write a function named filter, which will accept an array of int values, and an int named max. It should create and return another array of int values, where the new array contains all values from the original array that are less than or equal to max. For example, for the array

```
int[] test ={3,9,-2,6,1,8};
```

the function call

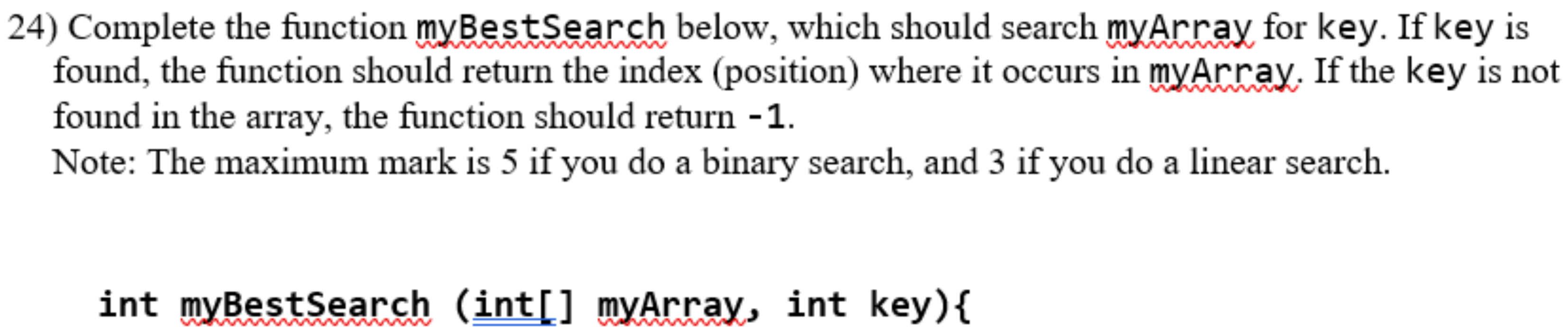
```
int[] result = filter(test,6);
```

should return the array

```
{3,-2,6,1}.
```

The array returned should be exactly the correct size to hold the required number of values. The original array may be any size, 0 or greater.

PROGRAMMING: PFA & SEARCHING



24) Complete the function myBestSearch below, which should search myArray for key. If key is found, the function should return the index (position) where it occurs in myArray. If the key is not found in the array, the function should return -1.

Note: The maximum mark is 5 if you do a binary search, and 3 if you do a linear search.

```
int myBestSearch (int[] myArray, int key){
```

BONUS EXTRA PROGRAMMING QUESTIONS

NO ANSWERS GIVEN FOR THESE – TRY THEM OUT YOURSELF!

SOME PRACTICE SUGGESTIONS FOR ARRAYS

- Write a function that:
  - takes in an int[ ], returns min value
  - takes in an int[ ], returns the average
  - takes in a String, returns if palindrome
  - takes in an array, returns the reverse array
    - modify in place (i.e. modify the original array) OR create a new array
  - merges two arrays into one large array
  - finds a value in an array & returns it
  - finds a character in a String & returns it
  - takes in two arrays & returns whether they are equal

PROGRAMMING PRACTICE: SCALING

- Write a program that draws a box at the centre of the canvas.
- The size of the box should depend on the mouse position.
  - Write one version of the program where the box is small when the mouse is at the top of the canvas and large when the mouse is at the bottom of the canvas.
  - Write a second version of the program where the box is large when the mouse is at the top of the canvas and small when the mouse is at the bottom of the canvas.

PROGRAMMING PRACTICE: REPEATING PATTERNS

- Write a program that prints a message to the console each time the user clicks the mouse on the canvas.
- The message printed should depend on a repeating pattern, as in
  “Hi” → “Yes” → “Bye” → “Hi” → “Yes” → …
- There should be several possible messages (at least 3).
- The next message in the pattern should be printed after the next click.

PROGRAMMING PRACTICE: COMBINE ARRAYS

- Write a function that will accept two arrays and return a third array.
- The arrays passed will be the same size.
- The array returned should contain all of the items from both input arrays.
- The items in the returned array should alternately be added from the input arrays.
- For example, if a = {3, 6, 9, 4} and b = {5, 0, 7, 2} are passed to the function, the returned array should be {3, 5, 6, 0, 9, 7, 4, 2}.
- This function might be called as in
  int[ ] c = combineArrays(a, b);