How to do well on the AP CS Free Response

2015 Questions

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions

M/C Review Question Banks

Live Programming Problems

Tons of great content!

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

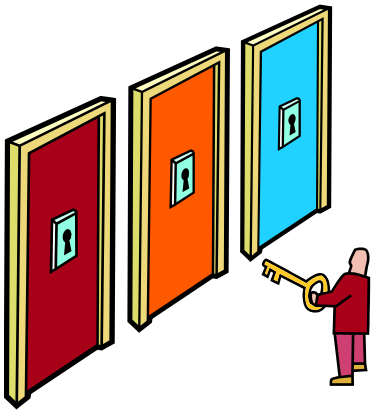
www.apluscompsci.com

Full Curriculum Solutions
Slides and Examples
Labs and Worksheets
Tests and Quizzes
Solutions provided for all

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com

AP Free Response Questions



© A+ Computer Science - www.apluscompsci.com

AP A Free Response Topics

Array / Array or Arrays

– for and for each loops, nested loops, array of arrays concepts

Make a Class - Basic

– create a basic class with instance variables, constructors, and methods

ArrayList of References / Strings

– get,set,remove,add,size – levels of abstraction

Make a Class - Interface / Abstract

– implement / extend – looks like this topic is back

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

-Read all 4 questions before writing anything

-answer the easiest question 1st

-most times question 1 is the easiest

-see if part B calls part A and so on

-many times part C consists of A and B calls

-write something on every question

-write legibly / use **PENCIL**!!!!!!!!!!

-keep track of your time

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

- When writing methods
  - use parameter types and names as provided
  - do not redefine the parameters listed
  - do not redefine the methods provided
  - return from all return methods
  - return correct data type from return methods

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

-When writing a class or methods for a class

  -know which methods you have

  -know which instance variables you have

  -check for public/private on methods/variables

  -return from all return methods

  -return correct data type from return methods

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

- When extending a class
  - know which methods the parent contains
  - have the original class where you can see it
  - make sure you have super calls
  - check for public/private on methods/variables
  - make super calls in sub class methods as needed

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

-When extending abstract / implementing interface

  -know which methods the parent contains

  -have the original class where you can see it

  -make sure you have super calls

  -check for public/private on methods/variables

  -make super calls in sub class methods as needed

  -implement all abstract methods in sub class

© A+ Computer Science - www.apluscompsci.com

AP A Free Response Topics

Array / Array or Arrays

– for and for each loops, nested loops, array of arrays concepts

Make a Class - Basic

– create a basic class with instance variables, constructors, and methods

ArrayList of References / Strings

– get,set,remove,add,size – levels of abstraction

Make a Class - Interface / Abstract

– implement / extend – looks like this topic is back

© A+ Computer Science - www.apluscompsci.com

Arrays and Array of Arrays

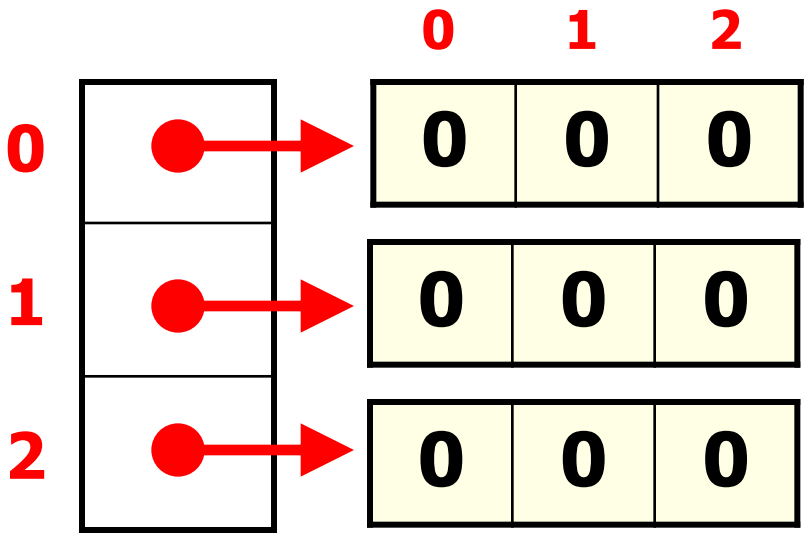
One question on the A test free response will require you to manipulate an array and / or an array of arrays.

© A+ Computer Science - www.apluscompsci.com

What is a matrix?

A matrix is an array of arrays.

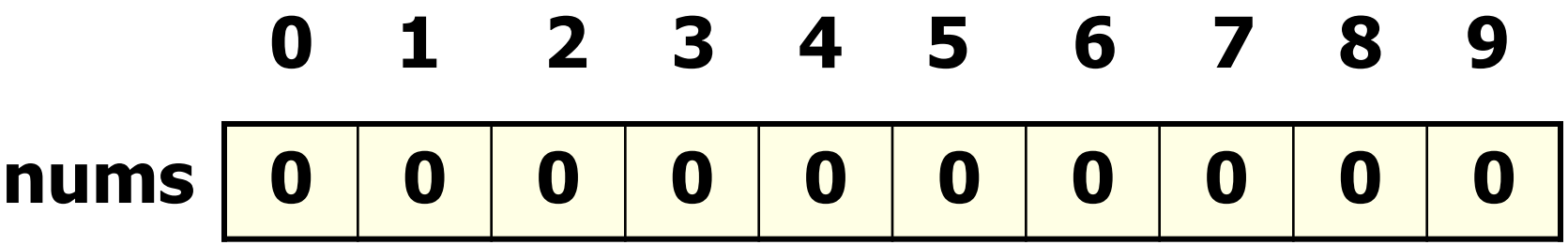
```
int[][] mat = new int[3][3];
```



© A+ Computer Science - www.apluscompsci.com

AP A - Arrays

```
int[] nums = new int[10];    //Java int array
```

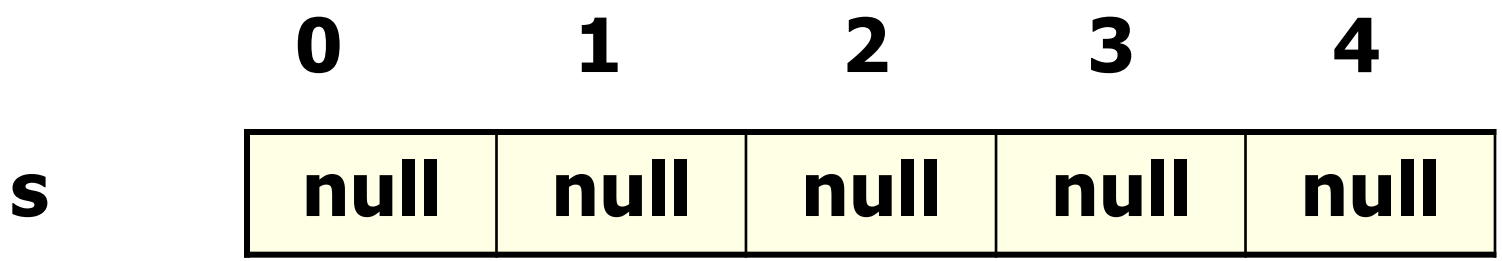


An array is a group of items all of the same type which are accessed through a single identifier.

© A+ Computer Science - www.apluscompsci.com

AP A - Arrays

```
String[] s = new String[5];
```



Arrays are automatically filled with values when instantiated.

© A+ Computer Science - www.apluscompsci.com

Matrix Variables

```
int size = 40;

int[][] mat = {{5,7,9,2},
               {5,3,4,6},
               {7,0,8,9}};

int[][] intMat = new int[size][size];
        //intMat is filled with zeros - 0s
```

© A+ Computer Science - www.apluscompsci.com

Matrix Variables

```
String[][] words = new String[4][4];
        //words is filled with null


double[][] dMat = new double[3][3];
        //dMat is filled with 0.0
```

© A+ Computer Science - www.apluscompsci.com

Printing Spots

```
int[][] mat = {{5,7,9,2,1,9},
               {5,3,4},
               {3,7,0,8,9}};

out.println(mat[2][1]);
out.println(mat[1][2]);
out.println(mat[0][3]);
out.println(mat[2][4]);
```

OUTPUT

```
7
4
2
9
```

© A+ Computer Science - www.apluscompsci.com

Printing Spots

```
int[][] mat = {{5,7,9,2,1,9},
               {5,3,4},
               {3,7,0,8,9}};

out.println(mat[7/4][0]);
out.println(mat[1*2][2]);
out.println(mat.length);
out.println(mat[0].length);
```

OUTPUT

5
0
3
6

© A+ Computer Science - www.apluscompsci.com

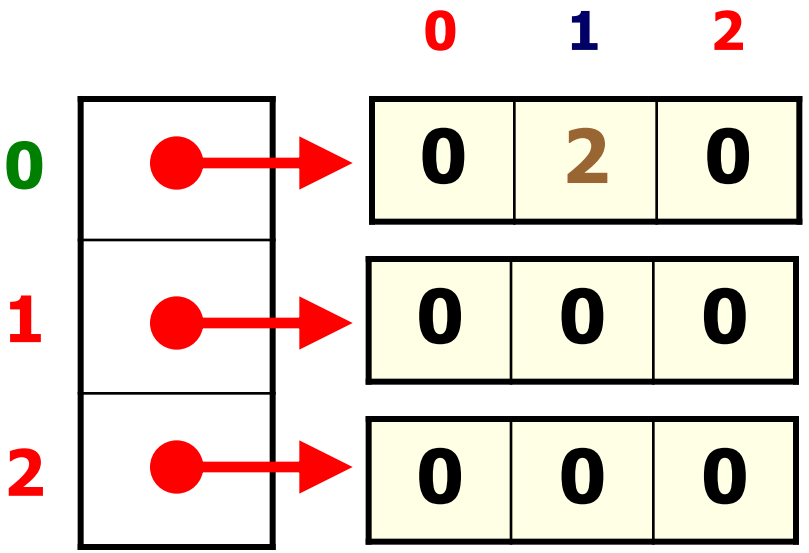
What is a matrix?

A matrix is an array of arrays.

```
int[][] mat = new int[3][3];
mat[0][1]=2;
```

Which array?

Which spot?

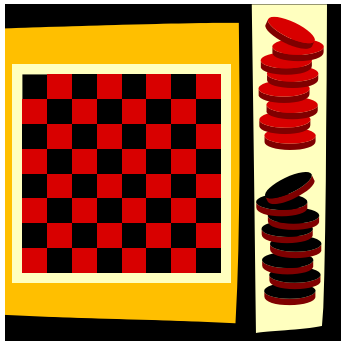


© A+ Computer Science - www.apluscompsci.com

Assigning Matrix Values

	0	1	2	3	4
0	0	0	0	5	0
1	0	0	0	0	0
2	0	0	7	0	0
3	0	0	0	0	0
4	0	3	0	0	0

```
mat[2][2]=7;
mat[0][3]=5;
mat[4][1]=3
```



© A+ Computer Science - www.apluscompsci.com

Assigning Matrix Values

```
for( int r = 0; r < mat.length; r++)
{
  for( int c = 0; c < mat[r].length; c++)
  {
       mat[r][c] = r*c;
  }
}
```

if mat was 3x3

0	0	0
0	1	2
0	2	4

© A+ Computer Science - www.apluscompsci.com

Nested Loop Review

```
int outer=1;
      //start      //stop      //increment
for(outer=1;  outer<=2;     outer++)
{
        //start      //stop    //increment
    for(int inner=1;  inner<=2;   inner++)
      out.println(outer + " " + inner);
    out.println();
}
```

OUTPUT

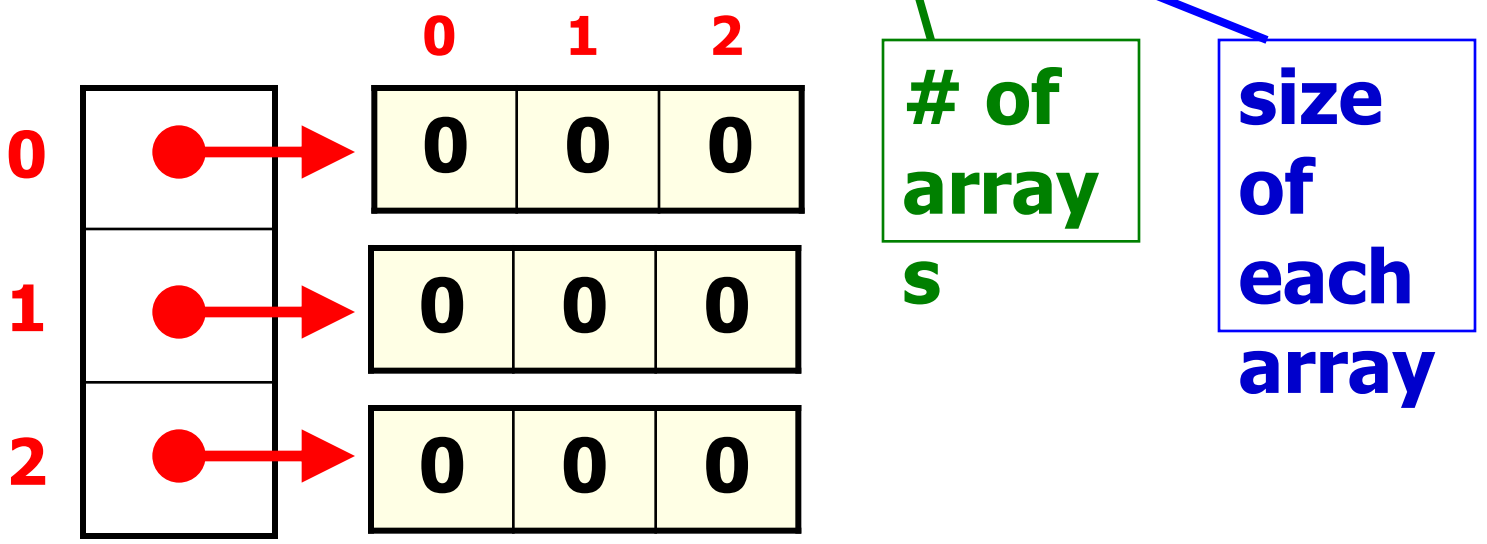
```
1 1
1 2

2 1
2 2
```

© A+ Computer Science - www.apluscompsci.com

What is a matrix?

A matrix is an array of arrays.

```
int[][] mat = new int[3][3];
```



© A+ Computer Science - www.apluscompsci.com

Printing an Array

```
int[][] mat = {{5,7},{5,3,4,6},{0,8,9}};

out.println(Arrays.toString(mat[0]));

out.println(Arrays.toString(mat[1]));
```

OUTPUT

```
[5, 7]
[5, 3, 4, 6]
```

© A+ Computer Science - www.apluscompsci.com

Printing an Array

```
int[] nums = {1,2,3,4,5,6,7};
for(int r=0; r<nums.length; r++)
{
    out.println(nums[r]);
}
```

length returns the # of elements/items/spots in the array!!!

OUTPUT

```
1
2
3
4
5
6
7
```

© A+ Computer Science - www.apluscompsci.com

Printing a Matrix

```
int[][] mat = {{5,7},{5,3,4,6},{0,8,9}};

for(int r=0; r<mat.length; r++)
{
    for(int c=0; c<mat[r].length; c++)
    {
        out.print(mat[r][c]);
    }
    out.println();
}
```

OUTPUT

```
5 7
5 3 4 6
0 8 9
```

© A+ Computer Science - www.apluscompsci.com

Printing a Matrix

```
int[][] mat = {{5,7},{5,3,4,6},{0,8,9}};

for( int[] row : mat )
{
  for( int num : row )
  {
    System.out.print( num + " ");
  }
  System.out.println();
}
```

OUTPUT

```
5 7
5 3 4 6
0 8 9
```

© A+ Computer Science - www.apluscompsci.com

2015 AP A

Free Response

© A+ Computer Science - www.apluscompsci.com

```
public static int arraySum( int[] arr )
{
    int sum = 0;
    for( int item : arr )
        sum = sum + item;
    return sum;
}
```

2015
Question 1
part A

```
public static int[] rowSums( int[][] arr2D )
{
  int[] ret = new int[arr2D.length];
  for( int i = 0; i < ret.length; i++ )
      ret[i] = arraySum(arr2D[i]);
  return ret;
}
```

2015 Question 1 part B

```
public static boolean isDiverse( int[][] arr2D )
{
   int[] totals = rowSums( arr2D );
   for( int i = 0; i < totals.length-1; i++ )
   {
       for( int j = i+1; j < totals.length; j++)
          if( totals[i] == totals[j] )
               return false;
   }
   return true;
}
```

This version is kinda lame, but I assume most students will do something like this.

2015 Question 1 part C – ver 1

```
public static boolean isDiverse( int[][] arr2D )
{
  int[] totals = rowSums( arr2D );
  ArrayList<Integer> used;
  used = new ArrayList<Integer>();
  for( int item : totals )
  {
     if( used.contains( item) )
        return false;
     else
        used.add( item );
   }
   return true;
}
```

2015 Question 1 part C – ver 2

This algorithm is more interesting and more of what I would expect from my students that participate in contests.

Provided by
A+ Computer Science

A+ Lab Assignments that cover the same matrix concepts as FR question # 1 ::

- **Magic Square and Spiral Matrix**
- **Forest and Tic-Tac-Toe**
- **Codes and Ciphers**

www.apluscompsci.com

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions

M/C Review Question Banks

Live Programming Problems

Tons of great content!

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com

AP A
Make a Class

One question on the A test free response will require that you create a class sometimes using inheritance and sometimes not.

You will need to create instance variables, constructors, and methods.

The 2015 question was a class question that involved strings.

© A+ Computer Science - www.apluscompsci.com

Class Components

© A+ Computer Science - www.apluscompsci.com

classes

```
public class Triangle
{
   private int sideA;
   private int sideB;
   private int sideC;
```

Instance variables

Instance variables store the state information for an object.

© A+ Computer Science - www.apluscompsci.com

constructors

```
public Triangle(int a, int b, int c)
{
   sideA=a;
   sideB=b;
   sideC=c;
}
```

Initialization Constructor

Constructors are similar to methods. Constructors set the properties of an object to an initial state.

© A+ Computer Science - www.apluscompsci.com

modifier methods

```
public void setSideA(int a )
{
    sideA=a;
}
```

Modifier methods are methods that change the properties of an object.

© A+ Computer Science - www.apluscompsci.com

accessor methods

```
public int getSideA()
{
  return sideA;
}
```

Accessor methods are methods that retrieve or grant access to the properties of an object, but do not make any changes.

© A+ Computer Science - www.apluscompsci.com

String

© A+ Computer Science - www.apluscompsci.com

String

frequently used methods

Name	Use
substring(x,y)	returns a section of the string from x to y not including y
substring(x)	returns a section of the string from x to length-1
length()	returns the # of chars
charAt(x)	returns the char at spot x
indexOf(c)	returns the loc of char c in the string, searching from spot 0 to spot length-1
lastIndexOf(c)	returns the loc of char c in the string, searching from spot length-1 to spot 0

© A+ Computer Science - www.apluscompsci.com

Strings

```
String sent = "alligators rule";
String find = "gato";

System.out.println( sent.indexOf( find ) );
System.out.println( sent.indexOf( "dog" ) );
System.out.println( sent.substring( 3 , 6 ) );
System.out.println( sent.substring( 6 ) );
```

OUTPUT

```
4
-1
iga
tors rule
```

© A+ Computer Science - www.apluscompsci.com

2015 AP A

Free Response

© A+ Computer Science - www.apluscompsci.com

2015 Question 2

```
public class HiddenWord
{
  private String word;

   public HiddenWord( String w )
   {
     word = w;
   }

   public String getHint( String guess )
   {
     String ret = "";
     for( int i = 0; i < word.length(); i++)
     {
           if( word.substring(i,i+1).equals(guess.substring(i,i+1)) )
             ret += word.substring( i, i+1);
           else if( word.indexOf( guess.substring(i,i+1))!=-1)
             ret += "+";
           else
             ret += "*";
     }
     return ret;
   }

}
```

Provided by A+ Computer Science

A+ Lab Assignments that cover the same string concepts as FR question # 2 ::

- **HexChecker, MorseCode**
- **Rock-Paper-Scissors**
- **Etch-A-Sketch**

www.apluscompsci.com

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions

M/C Review Question Banks

Live Programming Problems

Tons of great content!

www.facebook.com/APlusComputerScience

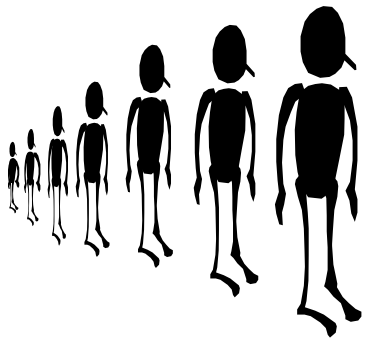
© A+ Computer Science - www.apluscompsci.com

ArrayList

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList

A typical ArrayList question involves putting something into an ArrayList and removing something from an ArrayList.



© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList

Arraylist is a class that houses an array.

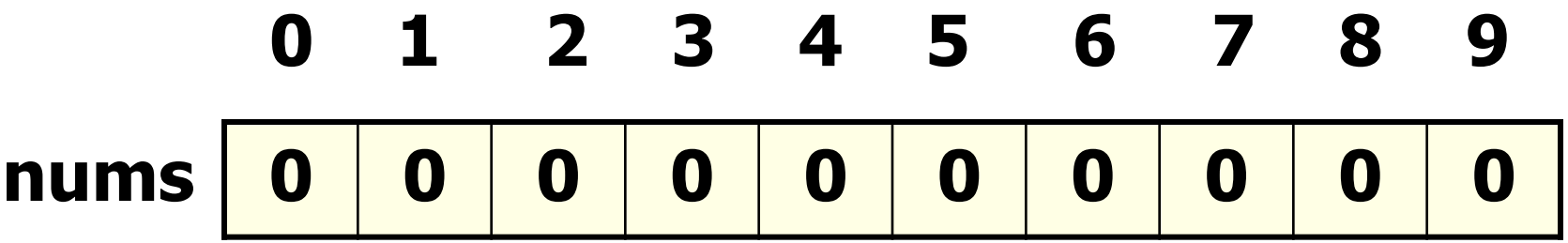
An ArrayList can store any type.

All ArrayLists store the first reference at spot / index position 0.

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList

```
int[] nums = new int[10];          //Java int array
```



An array is a group of items all of the same type which are accessed through a single identifier.

© A+ Computer Science - www.apluscompsci.com

ArrayList

frequently used methods

Name	Use
add(item)	adds item to the end of the list
add(spot,item)	adds item at spot – shifts items up->
set(spot,item)	put item at spot z[spot]=item
get(spot)	returns the item at spot return z[spot]
size()	returns the # of items in the list
remove()	removes an item from the list
clear()	removes all items from the list

```
import  java.util.ArrayList;
```

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList

```
List ray = new ArrayList();
ray.add("hello");
ray.add("whoot");
ray.add("contests");
out.println(((String)ray.get(0)).charAt(0));
out.println(((String)ray.get(2)).charAt(0));
```

OUTPUT

```
h
c
```

ray stores Object references.

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList

```
List<String> ray;
ray = new ArrayList<String>();
ray.add("hello");
ray.add("whoot");
ray.add("contests");
out.println(ray.get(0).charAt(0));
out.println(ray.get(2).charAt(0));
```

OUTPUT

```
h
c
```

ray stores String references.

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList Removal

```
int spot=list.size()-1;
while(spot>=0)
{

   if(list.get(spot).equals("killIt"))
      list.remove(spot);

   spot--;

}
```

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList Removal

```
for(int i=list.size()-1; i>=0; i--)
{

   if(list.get(i).equals("killIt"))
      list.remove(i);

}
```

© A+ Computer Science - www.apluscompsci.com

AP A - ArrayList Removal

```
int spot=0;
while(spot<list.size())
{
   if(list.get(spot).equals("killIt"))
      list.remove(spot);
   else
      spot++;
}
```

© A+ Computer Science - www.apluscompsci.com

2015 AP A

Free Response

© A+ Computer Science - www.apluscompsci.com

2015 Question 3 part A

```
public int getValueAt( int row , int col )
{
  for( SparseArrayEntry item : entries )
  {
    if( item.getRow()==row && item.getCol()==col)
      return item.getValue();
  }
  return 0;
}
```

You must know ArrayList!

2015 Question 3 - part B

```
public void removeCol( int col )
{
  for( int i = entries.size()-1; i>=0; i--)
 {
   SparseArrayEntry e = entries.get(i);
   if( e.getCol()==col)
   {
     entries.remove(i);
    }
   else if ( e.getCol()>col)
   {
     SparseArrayEntry up;
     up = new SparseArrayEntry( e.getRow(), e.getCol()-1, e.getValue() );
     entries.set( i, up);
   }
 }
 numCols--;
}
```

You must know ArrayList!

Provided by
A+ Computer Science

A+ Lab Assignments that cover the same list concepts as FR question # 3 ::

- **MadLib, WordList, Numbers, and Toys**
- **Moving Shapes**
- **BlackJack – 21**

www.apluscompsci.com

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions

M/C Review Question Banks

Live Programming Problems

Tons of great content!

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

A typical Abstract/Interface question requires that a class be written that extends the abstract class or implements the interface and that all abstract method(s) be implemented.

© A+ Computer Science - www.apluscompsci.com

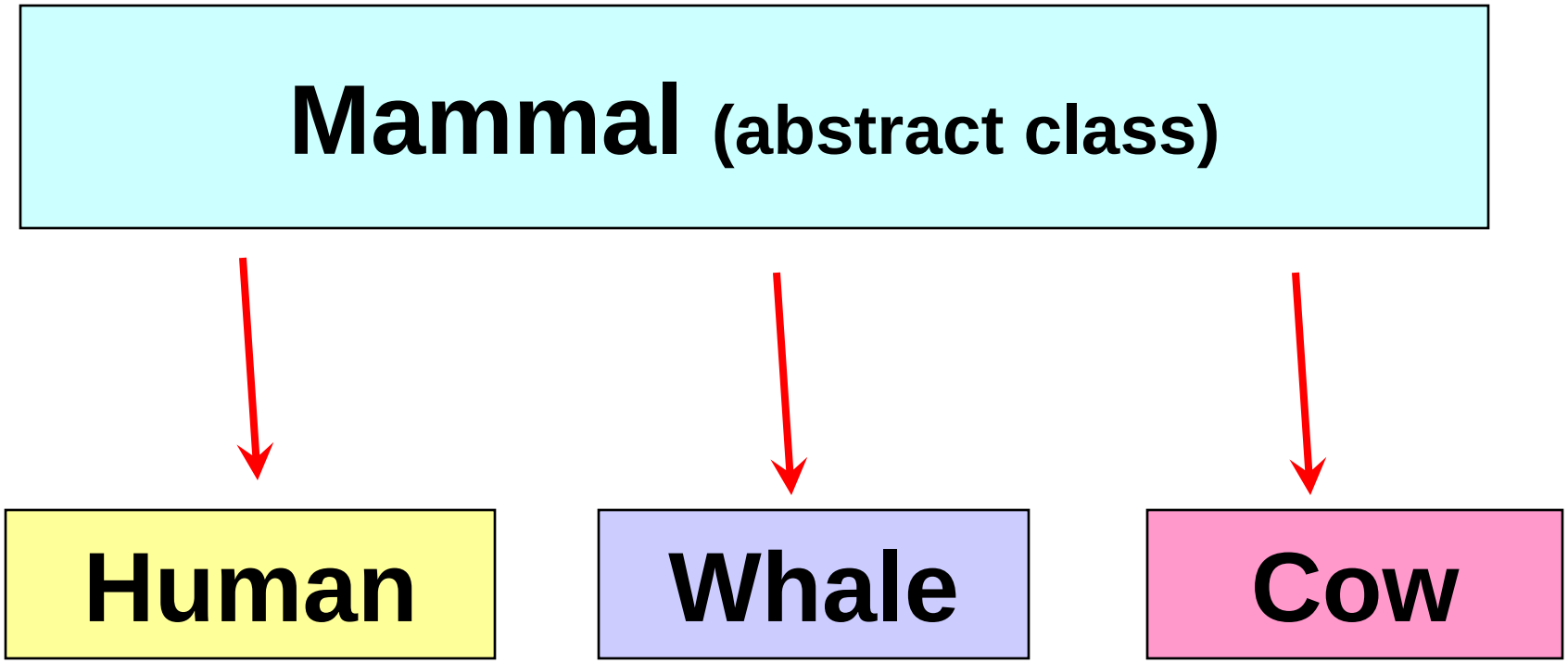
AP A - Abstract/Interface

Abstract classes are used to define a class that will be used only to build new classes.

No objects will ever be instantiated from an abstract class.

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface



© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

Any sub class that extends a super abstract class must implement all methods defined as abstract in the super class.

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract / Interface

```
abstract class Monster{
   private String myName;

   public Monster()   {
      myName ="";
   }

   public Monster(String name)   {
       myName =name;
   }

   public String getName()   {
      return myName;
   }

   public abstract void talk( );

   public String toString() {
      return myName + "\n";
   }
}
```

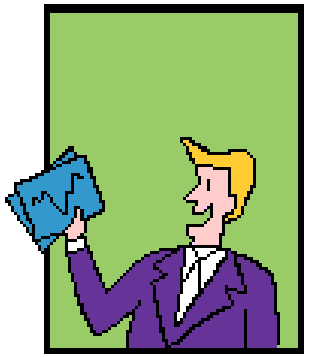
© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

Why define talk as abstract?

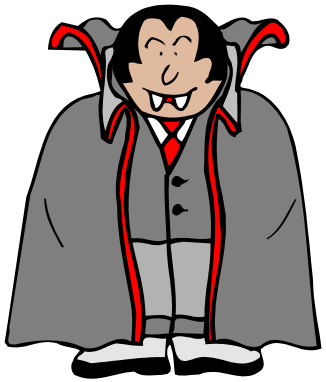
```
public abstract void talk( );
```

Does each Monster say the exact same thing?



© A+ Computer Science - www.apluscompsci.com

Vampire



```
class Vampire extends Monster
{
  public Vampire( String name )
  {
    super(name);
  }

  public void talk()
  {
    out.println("Vampire " + getName() +
                                    " say \"I want to drink your blood!\"");
  }
}
```

Sub Class

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

```
public abstract class APlus
{
  public APlus(int x)
    //constructor code not shown

  public abstract double goForIt();

  //other fields/methods not shown
}
```

Pet
Item

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

```
public class PassAPTest extends APlus
{
  public PassAPTest(int x)
  {
    super(x);
  }

  public double goForIt()
  {
    double run=0.0;
    //write some code   -   run = x*y/z
    return run;
  }

  //other fields/methods not shown
}
```

```
public abstract class APlus
{
  public APlus(int x)
    //constructor code not shown

  public abstract double goForIt();

  //other fields/methods not shown
}
```

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

```
public interface Exampleable
{
   int writeIt(Object o);
   int x = 123;
}
```

Methods are public abstract!
Variables are public static final!

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

```
public interface Exampleable
{
   public abstract int writeIt(Object o);
   public static final int x = 123;
}
```

Methods are public abstract!
Variables are public static final!

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

An interface is a list of abstract methods that must be implemented.

An interface may not contain any implemented methods.

Interfaces cannot have constructors!!!

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

Interfaces are typically used when you know what you want an Object to do, but do not know how it will be done.

If only the behavior is known, use an interface.

© A+ Computer Science - www.apluscompsci.com

AP A - Abstract/Interface

Abstract classes are typically used when you know what you want an Object to do and have a bit of an idea how it will be done.

If the behavior is known and some properties are known, use an abstract class.

© A+ Computer Science - www.apluscompsci.com

2015 AP A

Free Response

© A+ Computer Science - www.apluscompsci.com

```
public interface NumberGroup
{
  boolean contains( int val );
}
```

2015
Question 4
Part A

```
public class Range implements NumberGroup
{
  private int low;
  private int high;

  public Range( int lo, int hi)
  {
    low = lo;
    high = hi;
  }

  public boolean contains( int val )
  {
    return val >= low && val <= high;
  }
}
```

2015
Question 4
Part B

```
public class MultipleGroups implements NumberGroup
{
  private List<NumberGroup> groupList;

  //constructors , other methods, and such not shown

  public boolean contains( int val )
  {
    for( NumberGroup item : groupList )
    {
       if( item.contains( val) )
         return true;
     }
    return false;
  }
}
```

2015 Question 4 Part C

Provided by A+ Computer Science

A+ Lab Assignments that cover the same class concepts as FR question # 4 ::

- **Tic Tac Toe**
- **Lots of games**

www.apluscompsci.com

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions

M/C Review Question Banks

Live Programming Problems

Tons of great content!

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com

AP A Free Response Topics

Array / Array or Arrays

– for and for each loops, nested loops, array of arrays concepts

Make a Class - Basic

– create a basic class with instance variables, constructors, and methods

ArrayList of References / Strings

– get,set,remove,add,size – levels of abstraction

Make a Class - Interface / Abstract

– implement / extend – looks like this topic is back

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

-Read all 4 questions before writing anything

-answer the easiest question 1st

-most times question 1 is the easiest

-see if part B calls part A and so on

-many times part C consists of A and B calls

-write something on every question

-write legibly / use **PENCIL**!!!!!!!!!!

-keep track of your time

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

- When writing methods
  - use parameter types and names as provided
  - do not redefine the parameters listed
  - do not redefine the methods provided
  - return from all return methods
  - return correct data type from return methods

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

-When writing a class or methods for a class

  -know which methods you have

  -know which instance variables you have

  -check for public/private on methods/variables

  -return from all return methods

  -return correct data type from return methods

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

- When extending a class
  - know which methods the parent contains
  - have the original class where you can see it
  - make sure you have super calls
  - check for public/private on methods/variables
  - make super calls in sub class methods as needed

© A+ Computer Science - www.apluscompsci.com

Free Response Strategies

-When extending abstract / implementing interface

  -know which methods the parent contains

  -have the original class where you can see it

  -make sure you have super calls

  -check for public/private on methods/variables

  -make super calls in sub class methods as needed

  -implement all abstract methods in sub class

© A+ Computer Science - www.apluscompsci.com

AP A Free Response Topics

Array / Array or Arrays

– for and for each loops, nested loops, array of arrays concepts

Make a Class - Basic

– create a basic class with instance variables, constructors, and methods

ArrayList of References / Strings

– get,set,remove,add,size – levels of abstraction

Make a Class - Interface / Abstract

– implement / extend – looks like this topic is back

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions
Slides and Examples
Labs and Worksheets
Tests and Quizzes
Solutions provided for all

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com

Provided by
A+ Computer Science

Visit us at

www.apluscompsci.com

Full Curriculum Solutions

M/C Review Question Banks

Live Programming Problems

Tons of great content!

www.facebook.com/APlusComputerScience

© A+ Computer Science - www.apluscompsci.com