**[Q1](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2167" \o "Q1): Find out if the given number is an Armstrong number or not.**

**Description :-**

**An Armstrong number of three digits is an integer, where the sum of the cubes of its digits is equal to the number itself.**

**Consider the example: 371=> 3^3 + 7^3 + 1^3 = 371 ( If you add those all numbers, the final digit should be same as given number ).**

**public** **class** Armstrongnumber {

**public** **static** **void** main(String[] args) {

**int** number = 371, originalNumber, remainder, result = 0;

originalNumber = number;

**while** (originalNumber != 0)

{

remainder = originalNumber % 10;

result += Math.*pow*(remainder, 3);

originalNumber /= 10;

}

**if**(result == number)

System.***out***.println(number + " is an Armstrong number.");

**else**

System.***out***.println(number + " is not an Armstrong number.");

}

}

[**Q2**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2156)**. Find out all the Armstrong numbers falling in the range of 100-999**

**Description :-**

**An Armstrong number of three digits is an integer, where the sum of the cubes of its digits is equal to the number itself.**

**Consider the example: 371=> 3^3 + 7^3 + 1^3 = 371 ( If you add those all numbers, the final digit should be same as given number ).**

**Find the Armstrong numbers between 100 to 999.**

**public** **class** Armstrongnumber {

**public** **static** **void** main (String [] args)

{

**for** (**int** k = 100 ; k <= 1000 ; k++)

{

**int** n = k;

**int** d = 0;

**int** s = 0;

**while** (n > 0)

{

d = n % 10;

s = s + (d \* d \* d);

n = n / 10;

}

**if** (k == s)

{

System.***out***.println (k + " is Armstrong number");

}

}

}

}

[**Q3**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2173)**. Find out the simple as well as the compound interest of supplied value**

**Description:-**

**Simple Interest:-**Generally, simple interest paid or received over a certain period is a fixed percentage of the principal amount that was borrowed or lent

              Simple Interest = (P×r×n)/100

              where:

              P            =            Principal amount

              r             =            Annual interest rate

              n            =            Term of loan, in years

​**Compound Interest:-**Compound interest accrues and is added to the accumulated interest of previous periods; it includes interest on interest, in other words.

              Compound Interest = P(1+r)^t-P

              Where:

              P=Principal amount

              r=Annual interest rate

              t=Number of years interest is applied

**import** java.util .\*;

**class** sici

{

**public** **static** **void** main (String args[ ])

{

**double** pr, rate, t, sim,com;

Scanner sc=**new** Scanner (System. ***in***);

System.***out***.println("Enter the amount:");

pr=sc.nextDouble();

System. ***out***. println("Enter the No.of years:");

t=sc.nextDouble();

System. ***out***. println("Enter the Rate of interest");

rate=sc.nextDouble();

sim=(pr \* t \* rate)/100;

com=pr \* Math.*pow*(1.0+rate/100.0,t) - pr;

System.***out***.println("Simple Interest="+sim);

System.***out***. println("Compound Interest="+com);

}

}

[**Q4**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2262)**. Supply marks of three subject and declare the result, result declaration is based on below conditions:**

**Condition 1: -All subjects marks is greater than 60 is Passed**

**Condition 2: -Any two subjects marks are greater than 60 is Promoted**

**Condition 3: -Any one subject mark is greater than 60 or all subjects’ marks less than 60 is failed.**

**Description:-**

Specify the marks of 3 subjects and the results will be declared based on the conditions above and for reference go through the test cases for better understanding.

**import** java.util.Scanner;

**class** result{

**public** **static** String declareResults(**double** subject1Marks, **double** subject2Marks, **double** subject3Marks) {

String results = "faled";

**if**((subject1Marks>60) || (subject2Marks>60) || (subject3Marks>60)){

results = ("failed");

}

**else** **if**(((subject1Marks+subject2Marks)>60) || ((subject2Marks+subject3Marks)>60 )|| ((subject3Marks+subject1Marks)>60)){

results=("Passed"+"\n"+"promoted");

}

**else** **if**((subject1Marks+subject2Marks+subject3Marks)>60){

results= ("passed");

}

**return** results;

}

}

**public** **class** Q4{

**public** **static** **void** main(String[] args) {

Scanner sc= **new** Scanner(System.***in***);

System.***out***.println("Please enter Subject1 marks");

**double** subject1marks = sc.nextDouble();

System.***out***.println("Please enter Subject2 marks");

**double** subject2marks = sc.nextDouble();

System.***out***.println("Please enter Subject3 marks");

**double** subject3marks = sc.nextDouble();

System.***out***.print(result.*declareResults*(subject1marks,subject2marks,subject3marks));

}

}

[**Q5**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2153)**. Calculate the income tax on the basis of following table.**

**Note:-Assume slab is consider for Male, Female as well as Senior citizen**

**Slab                                   Income Range                             Tax payable in Percentage**

**Slab A                               0-1,80,000                                                  Nil**

**Slab B                               1,81,001-3,00,000                                      10%**

**Slab C                               3,00,001-5,00,000                                       20%**

**Slab D                               5,00,001-10,00,000                                     30%**

**Accept CTC from user and display tax amount**

**Description:-**

Given 4 different types of slabs along with the percentage of tax payable in association with income ranges which are applicalble to Male,Female as well as Senior citizen.You need to specify the CTC to display the taxable amount using the above slab rates.

**import** java.util.Scanner;

**class** TaxAmount{

**public** **static** **double** calculateTaxAmount(**int** ctc){

**double** tax = 0;

**if**(ctc<=80000){

tax = 0;

}

**else** **if**(ctc>=181001 && ctc<=300000){

tax = (ctc/100)\*10;

}

**else** **if**(ctc>=300001 && ctc<=500000){

tax = (ctc/100)\*20;

}

**else** **if**(ctc>=500001&& ctc<=1000000){

tax = (ctc/100)\*30;

}

**return** tax;

}

}

**public** **class** Q5 {

**public** **static** **void** main(String[] args) {

Scanner sc= **new** Scanner(System.***in***);

System.***out***.print("Please enter Your Ctc");

**int** ctc = sc.nextInt();

System.***out***.print("your income tax is :"+TaxAmount.*calculateTaxAmount*(ctc));

}

}

[**Q6**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2170)**. Consider a CUI based application, where you are asking a user to enter his Login name and password, after entering the valid user-id and password it will print the message “Welcome” along with user name. As per the validation is concerned, the program should keep a track of login attempts. After three attempts a message should be flashed saying “Contact Admin” and the program should terminate.**

**Description:-**

You have to create a CUI based application in which you will declare and initialize the userId and password.You have to enter the credentials when you login to the application(when you run the program it should ask the user to enter the credentials),if the entered credentials are validated correctly you should see a welcome message with the userId or else you should have total 3 attempts to enter the correct credentials.If you fail to enter the right credentials in your 3rd attempt you should display a message "Contact Admin".

**Test Cases:-**

**SampleInput:- (as per Specifications)**

userId = "Ajay",password = "password";

**import** java.util.Scanner;

**class** Login{

**static** String *userId* = "Ajay";

**static** String *password* = "password";

**static** **int** *i* = 3;

**public** **static** String loginUser(String user, String pass) {

*i*--;

String z="";

**if**((user.equals(*userId*) && pass.equals(*password*))) {

**return** "Welcome "+*userId*;

}

**else** **if**(*i*>=1) {

System.***out***.println("You have entered wrong credentials ,please enter the right credentials");

Q6.*main*(**null**);

}

**else** {

**return** "have entered wrong credentials 3 times \nContact Admin";

}

**return** z;

}

}

**public** **class** Q6 {

**public** **static** **void** main(String[] args) {

@SuppressWarnings("resource")

Scanner scanner=**new** Scanner(System.***in***);

System.***out***.println("userId: ");

String user=scanner.next();

System.***out***.println("password: ");

String pass=scanner.next();

System.***out***.print(Login.*loginUser*(user,pass));

}

}

[**Q7**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2151)**. There is an Array which is of the size 15, which may or may not be sorted. You should write a program to accept a number and search if it in contained in the array**

**Example:**

**5 12 14 6 78 19 1 23 26 35 37 7 52 86 47**

**Value to be search is 19**

**class** searcharray{

**public** **static** **boolean** searchArray(**int**[] arr, **int** toCheckValue) {

**boolean** t = **false**;

**for** (**int** i : arr) {

**if** (i == toCheckValue) {

t = **true**;

**break**;

}

}

**return** t;

}

}

**public** **class** Q7 {

**public** **static** **void** main(String[] args) {

**int** arr[] = {5, 12, 14, 6, 78, 19, 1, 23, 26, 35, 37, 7, 52, 86, 47};

**int** valueToCheck = 19;

System.***out***.println(searcharray.*searchArray*(arr, valueToCheck));

}

}

[**Q8**](https://adapt.in.capgemini.com/mod/vpl/view.php?id=2168)**. Using the below table write method apply sorting using Bubble Sort.**

**Example:**

**5 12 14 6 78 19 1 23 26 35 37 7 52 86 47**

**Description:-**

Bubble sort is a simple sorting algorithm. This sorting algorithm is comparison-based algorithm in which each pair of adjacent elements is compared and the elements are swapped if they are not in order. This algorithm is not suitable for large data sets as its average and worst case complexity are of Ο(n2) where n is the number of items.

**Example-**

              Input :          5 12 14 6 78 19 1 23 26 35 37 7 52 86 47

              Output:       1 5 6 7 12 14 19 23 26 35 37 47 52 78 86

**import** java.util.Scanner;

**class** Bubblesort {

**public** **static** **void** main(String []args) {

**int** n, c, d, swap;

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("number of integers to sort");

n = in.nextInt();

**int** array[] = **new** **int**[n];

System.***out***.println("Enter " + n + " integers");

**for** (c = 0; c < n; c++)

array[c] = in.nextInt();

**for** (c = 0; c < ( n - 1 ); c++) {

**for** (d = 0; d < n - c - 1; d++) {

**if** (array[d] > array[d+1])

{

swap = array[d];

array[d] = array[d+1];

array[d+1] = swap;

}

}

}

System.***out***.println("Sorted list of numbers");

**for** (c = 0; c < n; c++)

System.***out***.println(array[c]);

}

}

**Q9. Accept the marks of three students for the subject say A, B, C. Find the total scored and the average in all the subjects. Also Find the Total and Average scored by students in each respective Subject.**

**Description:-**

Enter the marks of 3 students for subjects A,B,C. Find the total marks secured by respective student in all the subjects and also find the total and average scored by students subject wise.

**import** java.util.Arrays;

**import** java.util.Scanner;

**class** student {

**public** **static** **int** studentsTotalMarksInAllSubjects(**int** student1total, **int** student2total, **int** student3total) {

**return** student1total+student2total+student3total;

}

**public** **static** **double** studentsAverageMarksInAllSubjects(**int** student1avg, **int** student2avg, **int** student3avg) {

**return** *studentsTotalMarksInAllSubjects*( student1avg, student2avg, student3avg)/3;

}

**public** **static** **int** subjectATotalByStudents(**int**[] marks) {

**return** *Total*(marks);

}

**public** **static** **int** subjectBTotalByStudents(**int**[] marks) {

**return** *Total*(marks);

}

**public** **static** **int** subjectCTotalByStudents(**int**[] marks) {

**return** *Total*(marks);

}

**public** **static** **double** subjectAAverageByStudents(**int**[] marks) {

**return** (*subjectATotalByStudents*(marks))/3;

}

**public** **static** **double** subjectBAverageByStudents(**int**[] marks) {

**return** (*subjectBTotalByStudents*(marks))/3;

}

**public** **static** **double** subjectCAverageByStudents(**int**[] marks) {

**return** (*subjectCTotalByStudents*(marks))/3;

}

**public** **static** **int**[] MarksEntry() {

**int**[] student = **new** **int**[4];

Scanner sc = **new** Scanner(System.***in***);

**int** i = 1;

**while** (i <= 3) {

System.***out***.println("please enter the Student marks for subject "+i );

**int** temp = sc.nextInt();

student[i] = temp;

i++;

}

**return** student;

}

**public** **static** **int** Total(**int**[] marks){

**int** total = 0;

**int** i;

**for** (i = 0; i < marks.length; i++)

total += marks[i];

**return** total;

}

}

**public** **class** Assignment1Q9 {

**public** **static** **int**[] *SubjectA* = **new** **int**[4];

**public** **static** **int**[] *SubjectB* = **new** **int**[4];

**public** **static** **int**[] *SubjectC* = **new** **int**[4];

**public** **static** **int**[] *student1* = **new** **int**[4];

**public** **static** **int**[] *student2* = **new** **int**[4];

**public** **static** **int**[] *student3* = **new** **int**[4];

**public** **static** **void** main(String[] args) {

*student1* = student.*MarksEntry*();

*student2* = student.*MarksEntry*();

*student3* = student.*MarksEntry*();

*SubjectA* = **new** **int**[]{*student1*[1], *student2*[1], *student3*[1]};

*SubjectB* = **new** **int**[]{*student1*[2], *student2*[2], *student3*[2]};

*SubjectC* = **new** **int**[]{*student1*[3], *student2*[3], *student3*[3]};

**int** SubATotal = (student.*Total*(*SubjectA*));

**int** SubBTotal = (student.*Total*(*SubjectB*));

**int** SubCTotal = (student.*Total*(*SubjectC*));

**int** Student1Total = (student.*Total*(*student1*));

**int** Student2Total = (student.*Total*(*student2*));

**int** Student3Total = (student.*Total*(*student3*));

**double** SubjAAvg = (student.*subjectAAverageByStudents*(*SubjectA*));

**double** SubjBAvg = (student.*subjectBAverageByStudents*(*SubjectB*));

**double** SubjCAvg = (student.*subjectCAverageByStudents*(*SubjectC*));

**int** AllStudentsTotal = (student.*studentsTotalMarksInAllSubjects*(Student1Total,Student2Total,Student3Total));

**double** AllStudentsAvg = (student.*studentsAverageMarksInAllSubjects*(Student1Total,Student2Total,Student3Total));

System.***out***.println(AllStudentsTotal);

System.***out***.println(AllStudentsAvg);

System.***out***.println(SubATotal);

System.***out***.println(SubjAAvg);

System.***out***.println(SubBTotal);

System.***out***.println(SubjBAvg);

System.***out***.println(SubCTotal);

System.***out***.println(SubjCAvg);

}

}