**1**. Find the nth term of the series

0,5,26,17,124,37,342,65,728,101

Explanation: it is a mixed series

If you don’t find logic separate terms at even and odd places

On separating

🡺Odd places: 0, 26, 124, 342, 728

Here the series is 13-1, 33-1, 53-1, 73-1......... i.e. (odd numbers 3-1) up to n

🡺Even places: 5, 17, 37, 65, 101,

Here the series 22+1, 42+1, 62+1, 82+1,102+1, i.e. (even numbers2+1) up to n

It is easy to identify such series for those who has idea on squares and cubes up to 30 so must go through them which is also helpful for quant’s

Now it’s just to add the for loop

C code:

#include<stdio.h>

main()

{

int n;

printf("enter the lenght of series:");

scanf("%d",&n);

for(int i=1;i<=n;i++)

{

if(i%2==0)

printf("%d\t",(i\*i)+1);

else

printf("%d\t",(i\*i\*i)-1);

}

}

import java.util.Scanner;

public class test {

public static void main(String[] args) {

int n;

System.out.print("enter the lenght of series:");

n = STDIN\_SCANNER.nextInt();

for(int i = 1; i <= n; i++) {

if(i % 2 == 0) {

System.out.print((i \* i + 1) + "\t");

} else {

System.out.print((i \* i \* i - 1) + "\t");

}

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

Output:

C:\Users\DELL\Pictures\Screenshots\Screenshot (528).png

Sometimes they would ask only the term at the particular position

i.e. find the 15th term of the series in that case just display 15th term only as 15 is odd position the answer (odd numbers 3-1)=153-1=3375-1=3374

Just modify the code as

#include<stdio.h>

main()

{

int n;

printf("which term of the series you need:");

scanf("%d",&n);

for(int i=1;i<=n;i++)

{

if(i==n)

{

printf("%d\t",(i\*i\*i)-1);

}

}

}

import java.util.Scanner;

public class test {

public static void main(String[] args) {

int n;

System.out.print("which term of the series you need:");

n = STDIN\_SCANNER.nextInt();

for(int i = 1; i <= n; i++) {

if(i == n) {

System.out.print((i \* i \* i - 1) + "\t");

}

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

Out put

C:\Users\DELL\Pictures\Screenshots\Screenshot (533).png

2.generate the series 0,3,4,9,8,15,12,21,16,27,20,33,24,39 or find the 12th term of the series

By looking at the question u feel it’s very difficult but it’s very funny.

people that it is alternate multiples of 2 and 3 but it’s not

The series starts with 0

So 0,1,2,3,4,5,6

Separate odd and even numbers

Odd numbers 1 3 5 7

1\*3, 3\*3, 5\*3, 7\*3 will fetch you

3, 9, 15, 21

Even numbers 2, 4, 6, 8

2\*2, 4\*2, 6\*2, 8\*2 will fetch you

4, 8, 12, 16

If it is an odd number return the product with 3 if even return product with 2

As 12th is even just return 12\*2

As 13th is odd just return 13\*3 that why it is funny ☺

#include<stdio.h>

main()

{

int n;

printf("enter the lenght of series:");

scanf("%d",&n);

for(int i=0;i<=n;i++)

{

if(i%2==0)

printf("%d\t",i\*2);

else

printf("%d\t",i\*3);

}

}

import java.util.Scanner;

public class test {

public static void main(String[] args) {

int n;

System.out.print("enter the lenght of series:");

n = STDIN\_SCANNER.nextInt();

for(int i = 0; i <= n; i++) {

if(i % 2 == 0) {

System.out.print((i \* 2) + "\t");

} else {

System.out.print((i \* 3) + "\t");

}

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

Output

C:\Users\DELL\Pictures\Screenshots\Screenshot (535).png

3. Find the sum of digits of number until it becomes a single digit numbers

Ex: 7892

7+9+8+2=26

2+6=8

Always perform % and / by 10 for separating digits until it becomes 0

Step 1:

7892%10=2

7892/10=789

Step 2:

789%10=9

789/10=78

Step 3:

78%10=8

78/10=7

Step4:

7%10=7

7/10=0

As it become zero u can stop now

It can be generalised in a loop as

while (n!=0)

{

r=n%10;

n=n/10;

sum=sum+r;

}

We cannot stop here again we need to add the digits of sum and repeat this process until it becomes a single digit .so we will use **go to** and ***l*abel** concept

#include<stdio.h>

main()

{

int n,r,sum=0;

printf("enter a num:");

scanf("%d",&n);

repeat:while(n!=0)

{

r=n%10;

n=n/10;

sum=sum+r;

}

//printf("\n sum is %d",sum);

if(sum>9)//to check if it is a single digit or not?

{

n=sum;

sum=0;

goto repeat;

}

else

printf("\nsum is %d",sum);

}

Java:

import java.util.Scanner;

public class singlesumdigit{

public static void main(String[] args) {

int n = 0;

int r = 0;

int sum = 0;

final int posLoop = 1;

for(int pos = 0; true;) switch(pos) {

default:

System.out.print("enter a num:");

n = STDIN\_SCANNER.nextInt();

case posLoop:

while(n != 0) {

r = n % 10;

n = n / 10;

sum = sum + r;

}

// printf("\nsum is %d",sum);

if(sum > 9 /\* to check if it is a single digit or not \*/) {

n = sum;

sum = 0; // agiain start like a new number

pos = posLoop;

continue;

} else {

System.out.print("\nsum is " + sum);

}

return;

}

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

4.Accept 2 strings ,all letters at odd position of string 1 followed by all letters at even position at string2 should form a third string then merge all the three strings and display

Ex: India(Ida)

Pakisthan (aita)

Third string should be Idaaita

Final result is IndiaPakisthanIdaaita

Assume the string size is max of 100 for all the three

#include<stdio.h>

#include<string.h>

main()

{

char str1[100],str2[100],str3[100];

printf("enter string 1:");

gets(str1);

printf("enter string 2:");

gets(str2);

int j=0;

for(int i=0;str1[i]!='\0';i++)

{

if(i%2==0)

{

str3[j]=str1[i];

j++;

}

}

for(int i=0;str2[i]!='\0';i++)

{

if(i%2!=0)

{

str3[j]=str2[i];

j++;

}

}

puts(str3);

for(int i=0;str3[i]!='\0';i++)

{

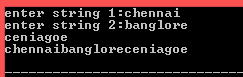
str3[i]=str3[i];

}

puts(strcat(strcat(str1,str2),str3));

}

Output:



Java Code:

public class test {

public static void main(String[] args) {

String8 str1 = new String8(100), str2 = new String8(100), str3 = new String8(100);

System.out.print("enter string 1:");

str1.copyFrom(STDIN\_SCANNER.nextLine());

System.out.print("enter string 2:");

str2.copyFrom(STDIN\_SCANNER.nextLine());

int j = 0;

for(int i = 0; str1.get(i) != '\0'; i++) {

if(i % 2 == 0) {

str3.set(j, str1.get(i));

j++;

}

}

for(int i = 0; str2.get(i) != '\0'; i++) {

if(i % 2 != 0) {

str3.set(j, str2.get(i));

j++;

}

}

System.out.println(str3);

for(int i = 0; str3.get(i) != '\0'; i++) {

str3.set(i, str3.get(i));

}

System.out.println(strcat(str1.copyFrom(str1 + "" + str2), str3));

}

public final static Scanner STDIN\_SCANNER = new Scanner(System.in);

}

☺