CHAITANYA BN

4AL17IS054

5TH SEM ISE

**5th August**

**Sum of Digits of a Five Digit Number**

**Task**

In this challenge, you have to input a five digit number and print the sum of digits of the number.

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main() {

int n;

scanf("%d", &n);

int digit, temp, sum = 0;

temp = n;

while(temp > 0)

{

digit = temp % 10;

sum = sum + digit;

temp = temp / 10;

}

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

return 0;

}

Output:16 Points:15

# Bitwise Operators

**Task**   
Given set , find: the maximum value of  which is less than a given integer , where  and  (where ) are two integers from set .

the maximum value of  which is less than a given integer , where  and  (where ) are two integers from set .

the maximum value of  which is less than a given integer , where  and  (where ) are two integers from set .

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

void calculate\_the\_maximum(int n, int k) {

int maxAnd = 0;

int maxOr = 0;

int maxXor = 0; for (int i=1; i<=n; i++) {

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

if (((i&j) > maxAnd) && ((i&j) < k)) {

maxAnd = i&j;

}

if (((i|j) > maxOr) && ((i|j) < k)) {

maxOr = i|j;

}

if (((i^j) > maxXor) && ((i^j) < k)) {

maxXor = i^j;

}

}

}printf("%d\n%d\n%d\n", maxAnd, maxOr, maxXor);

}

int main() {

int n, k;

scanf("%d %d", &n, &k);

calculate\_the\_maximum(n, k);

return 0;

}

Output: 2 3 3 Points:15

Conditional Statements in C

**Task**

Given a positive integer denoting , do the following:

* If ,1<=n<=9 then print the lowercase English word corresponding to the number (e.g., one for , two for , etc.).
* If ,n>9 print Greater than 9.

#include <stdio.h>

static const char \*strings[] = {"one","two","three","four","five",

"six","seven","eight","nine"};

int main()

{

int n = 0;

if (scanf("%d",&n) < 1)

return 1;

if (n >= 1 && n <= 9)

printf("%s",strings[n-1]);

else

printf("Greater than 9");

return 0;

}

Output: five Points:15

**Date:6th August**

# Calculate the Nth term

**Task**

There is a series, , where the next term is the sum of pervious three terms. Given the first three terms of the seriesa,b,c, , , and respectively, you have to output the nth term of the series using recursion.

Recursive method for calculating nth term is given below.

**Input Format**

* The first line contains a single integer,n .
* The next line contains 3 space-separated integers, a,b, , and c .

#include <stdio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

int find\_nth\_term(int n, int a, int b, int c) {

if(n == 1)

return a;

else if (n == 2)

return b;

else if (n == 3)

return c;

return find\_nth\_term

(n-1,a,b,c)+find\_nth\_term(n2,a,b,c)+find\_nth\_term(n-3,a,b,c);

} int main() {

int n, a, b, c;

scanf("%d %d %d %d", &n, &a, &b, &c);

int ans = find\_nth\_term(n, a, b, c);

printf("%d", ans);

return 0;

}

Output: 11 Points:15

# Sum and Difference of Two Numbers

**Task**

Your task is to take two numbers of int data type, two numbers of float data type as input and output their sum:

1. Declare  variables: two of type int and two of type float.
2. Read  lines of input from stdin (according to the sequence given in the 'Input Format' section below) and initialize your variables.
3. Use the  and  operator to perform the following operations:
   * Print the sum and difference of two int variable on a new line.
   * Print the sum and difference of two float variable rounded to one decimal place on a new line.

#include<stdio.h>

int main(){

int i, j;

float f, g;

scanf("%d %d %f %f", &i, &j, &f, &g);

printf("%d %d\n%.1f %.1f", i+j, i-j, f+g, f-g);

return 0;

}

* Output: 11 Points:5

# For Loop in C

For each integer n in the interval(a,b)  (given as input) :

* If 1<=n<=9, then print the English representation of it in lowercase. That is "one" for , "two" for , and so on.
* Else if n>9 and it is an even number, then print "even".
* Else if n>9 and it is an odd number, then print "odd".

#include <string.h>

#include <math.h>

#include <stdlib.h>

int main()

{ int a, b;

scanf("%d\n%d", &a, &b);

char labels[11][6] = {"one", "two", "three", "four","five", "six", "seven", "eight", "nine", "even", "odd"}int labels\_index;

for (int i=a; i<=b; i++) {

labels\_index = i <= 9 ? i - 1 : 9 + i % 2;

printf("%s\n", labels[labels\_index]);

}

return 0;

}

Output: eight nine even odd Points:10

# Simple Array Sum

Given an array of integers, find the sum of its elements.

For example, if the arrayar=[1,2,3] 1+2+3=6 , , so return 6.

#include <stdlib.h>

#include <string.h>

int main()

{

int sum=0,i,size,array[1000];

scanf("%d",&size);

for(i=size;i>=0;i--)

{

scanf("%d",&array[i]);

}

for(i=size;i>=0;i--)

{

sum+=array[i];

}

printf("%d",sum);

return 0;

}

Output: 31 Points:10