Write a C function – calc , to do simple integer arithmetic. Pass suitable function to calc to implement +, -, \*, and /

#include <stdio.h>

int calc(int (\*)(int,int), int, int); // First parameter - int (\*)(int,int) is a pointer to function which

// has two integer parameters and returns an int

int add(int,int);

int sub(int,int);

int mul(int,int);

int div(int,int);

int main()

{ int op; int n1, n2;int result;

printf("Enter 0 for add, 1 for sub, 2 for mul, 3 for div\n");

printf("Please enter code for arithmetic op\n");

scanf("%d", &op);

printf("Please enter two operands\n");

scanf("%d %d", &n1,&n2);

switch (op)

{ case 0: result = calc(add,n1,n2); // add is argument corresponding to function pointer

break;

case 1: result = calc(sub,n1,n2);

break;

case 2: result = calc(mul,n1,n2);

break;

case 3: result = calc(div,n1,n2);

break;

default: printf("Invalid op code and ignore results\n");

break;

}

printf("Result = %d\n", result);

return 0;

}

/\* First parameter func is pointer to function \*/

int calc (int(\*func)(int , int ), int m,int n)

{ int res;

res = (\*func)(m,n); // dereferencing func => call or invoke function

return res;

}

int add (int m, int n)

{ return m+n;

}

int sub (int m, int n)

{ return m-n;

}

int mul (int m, int n)

{ return m\*n;

}

int div (int m, int n)

{ return m/n;

}

Notes - Problem statement is chosen for the purpose of showing a simple example for passing a function to function

Array of function pointer method can also be used to solve this