Assignment 1

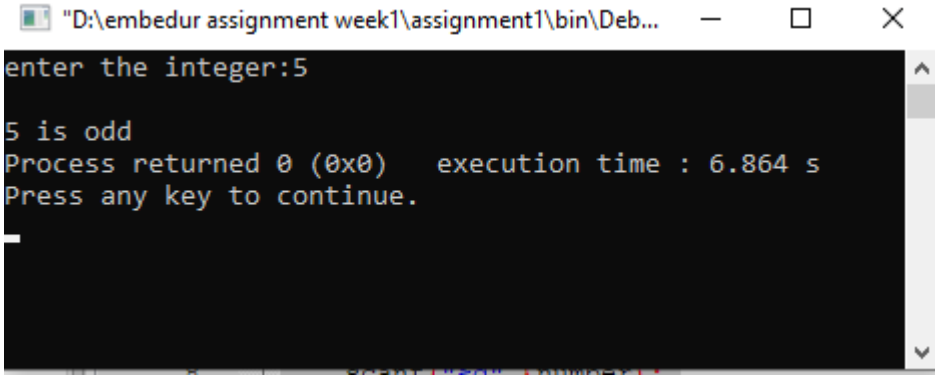
1. Write a C program to determine the given number is odd or even using Bitwise operators

CODE:

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
   int number;
   printf("enter the integer:");
   scanf("%d",&number);
   if(number&1)
      printf("\n%d is odd",number);
   else
      printf("\n%d is even",number);
   return 0;
}
```

OUTPUT:



2. Write a C program to count the number of bits set in a number.

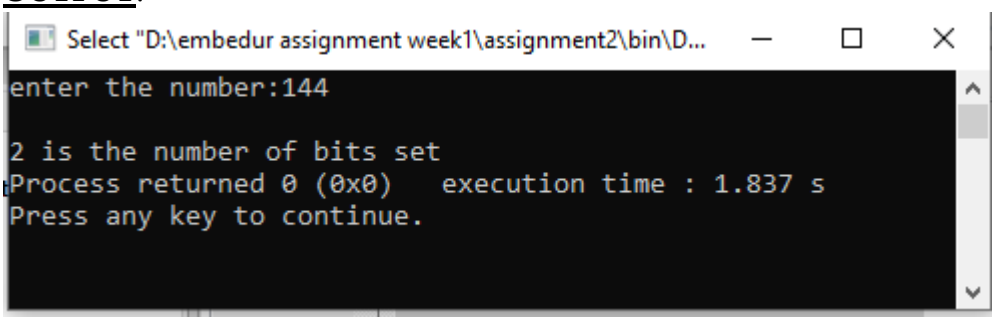
CODE:

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
   int n,count=0;
   printf("enter the number:");
   scanf("%d",&n);
   for(int i=0;i<32;i++)
   {
      int shift=n>>i;
```

```
        if(shift&1)
        {
           count=count+1;
        }
    }
    printf("\n%d is the number of bits set",count);
    return 0;
}
```

OUTPUT:



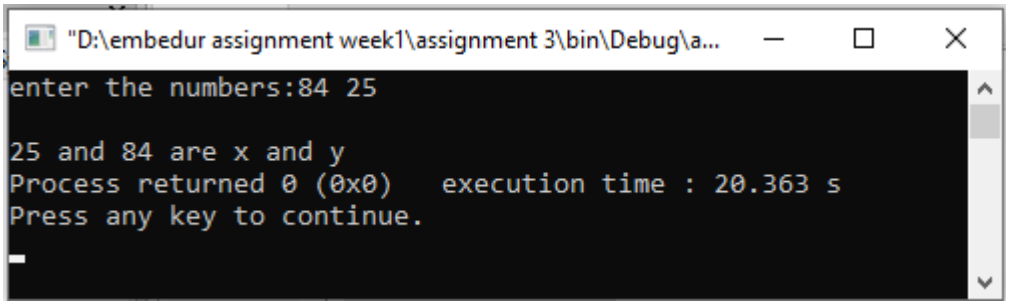
3. Write a C program to swap two numbers. Use a function pointer to do this operation.

CODE:

```
#include <stdio.h>
#include <stdlib.h>
int swap(int* a,int* b)
    {
       int temp;
       temp=*a;
       *a=*b;
       *b=temp;
    }

int main()
{
    int x,y;
    int (*fp)(int*,int*);
    printf("enter the numbers:");
    scanf("%d %d",&x,&y);
    fp=swap;
    (*fp)(&x,&y);
    printf("\n%d and %d are x and y",x,y);
    return 0;
}
```

OUTPUT:



4. . Write an equivalent pointer expression for fetching the value of array element a[i][j][k][2]

   ANSWER: *(*(*(a+i)+j)+k)

5. Write a C program to Multiply two matrix (n*n) using pointers.

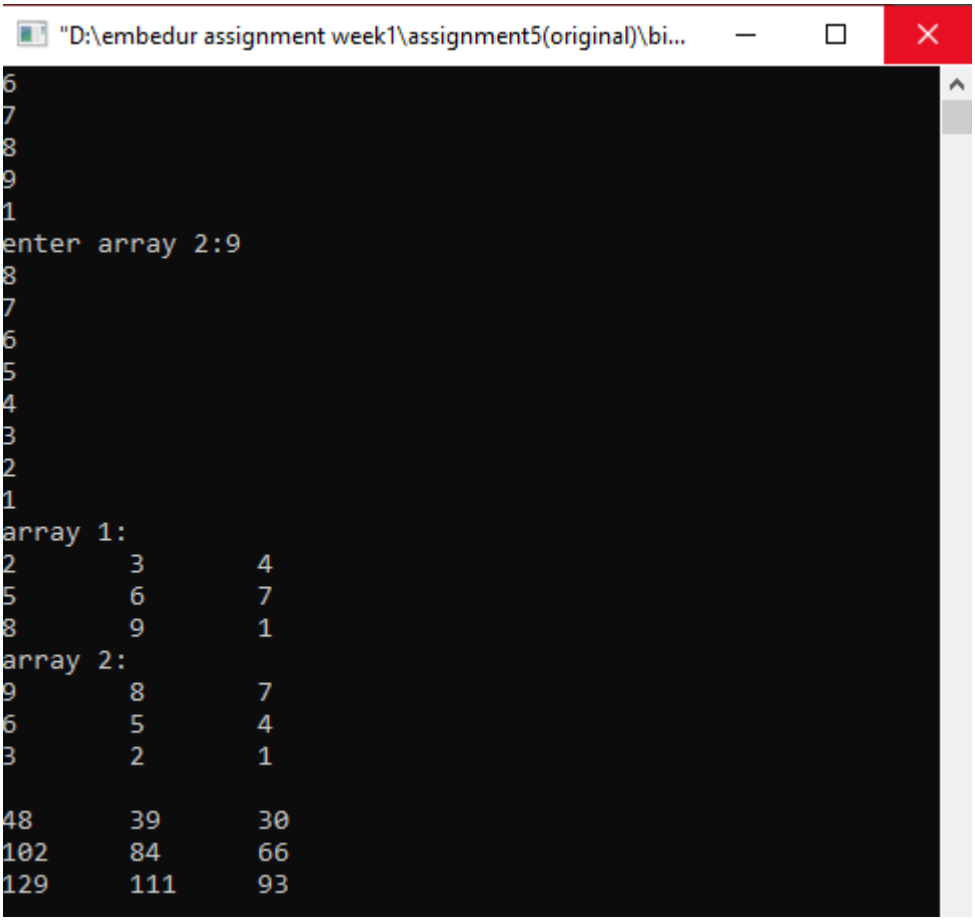
   CODE:

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
   int a1[3][3],a2[3][3],res[3][3];
   printf("enter array 1:");
   for(int i=0;i<3;i++)
   {
      for(int j=0;j<3;j++)
         scanf("%d",(*(a1+i)+j));
   }
   printf("enter array 2:");
   for(int i=0;i<3;i++)
   {
      for(int j=0;j<3;j++)
         scanf("%d",(*(a2+i)+j));
   }
   printf("array 1:\n");
   for(int i=0;i<3;i++)
   {
      for(int j=0;j<3;j++)
         printf("%d\t",*(*(a1+i)+j));
      printf("\n");
   }
   printf("array 2:\n");
   for(int i=0;i<3;i++)
   {
      for(int j=0;j<3;j++)
         printf("%d\t",*(*(a2+i)+j));
```

```
        printf("\n");
    }
    printf("\n");
    multiply(a1,a2);
    return 0;
}
void multiply(int a1[][3],int a2[][3])
{
    int mul[3][3],sum;
    for(int i=0;i<3;i++)
    {
        for(int j=0;j<3;j++)
        {
            sum=0;
            for(int k=0;k<3;k++)
            {
                sum+=(*(*(a1+i)+k))*(*(*(a2+k)+j));
            }
            *(*(mul+i)+j)=sum;
        }
    }
    for(int l=0;l<3;l++)
    {
        for(int m=0;m<3;m++)
        {
                printf("%d\t",mul[l][m]);
        }
        printf("\n");
    }
}
```

OUTPUT:



6. Find the output of the following // Consider the compiler is 32-bit machine

```
#include <stdio.h>
typedef struct
{
int A;
char B;
char C;
} InfoData;
int main(int argc, char *argv[])
{
//Calculate size of structure
printf("\n Size of Structure = %d\n\n",sizeof(InfoData));
return 0;
}
```

ANSWER: size of structure is 8

7. Find the output of the following // Consider the compiler is 32-bit machine

```
#include <stdio.h>
typedef struct
{
char A;
double B;
char C;
} InfoData;
int main(int argc, char *argv[])
{
//Calculate size of structure
printf("\n Size of Structure = %d\n\n",sizeof(InfoData));
return 0;
}
```

ANSWER: size of structure is 24

8. Find the output of the following // Consider the compiler is 32-bit machine

```
#include <stdio.h>
#include <stdint.h>
int main()
{
   unsigned int  var = 0x12345678;
   unsigned int  rev = 0;
   for (int j = 0; j < 8; j++)
   {
      rev = (rev << 4) | ((var >> (4 * j)) & 0xF);
   }
   printf("%X", rev);
   return 0;
}
```

ANSWER: 87654321