

## CSE 115 Lab on simple loop (part 3)

**1. Write a C program to read 2 integers: n and r from user and compute the value of  ${}^nP_r = \frac{n!}{(n-r)!}$**

Inefficient Code	Efficient Code
<pre>#include &lt;stdio.h&gt;  void main() {     int n,r,m=1,d=1,i;     printf("Enter n and r: ");     scanf("%d %d", &amp;n, &amp;r);     //compute n!     for(i=1;i&lt;=n;i++)         m*=i;      //compute (n-r)!     for(i=1;i&lt;=n-r;i++)         d*=i;      printf("nPr = %d", m/d); }</pre>	<pre>#include &lt;stdio.h&gt;  void main() {     int n,r,p=1,i;      printf("Enter n and r: ");     scanf("%d %d", &amp;n, &amp;r);      //compute nPr = (n-r+1)(n-r+2)... (n-1)n     for(i=n-r+1;i&lt;=n;i++)         p*=i;      printf("nPr = %d", p); }</pre>

**Try Yourself 1: Write a C program to read 2 integers: n and r from user and compute the value of  ${}^nC_r = \frac{n!}{r!(n-r)!}$**

**2. Write a C program to find Least Common Multiple (LCM) of two given numbers.**

```
#include <stdio.h>

void main()
{
    int i, n1, n2, max, lcm=1;

    printf("Enter any two numbers to find LCM: ");
    scanf("%d %d", &n1, &n2);

    i = max = (n1>n2) ? n1 : n2; //compute the max of n1 and n2; this is the lowest possible value of LCM
    while(1) //while condition is always true (1)
    {
        if(i%n1==0 && i%n2==0)    // If i is a multiple of both n1 and n2 then i is the LCM of n1 and n2
        {
            lcm = i;
            break; //break out of the loop since LCM has been found
        }
        //we come to the next line if current value of i is not the LCM
        i += max; //values of i are: max, 2*max, 3*max, ..., all of which are possible values of LCM
    }

    printf("\nLCM of %d and %d = %d\n", n1, n2, lcm);
}
```

**Try Yourself 2: Write a C program to read 3 integers and compute their LCM.**

### 3. Write a C program display a given number in words starting from its rightmost digit

<pre>#include &lt;stdio.h&gt; void main() {     int num, i;     printf("Enter any number to print in words: ");     scanf("%d", &amp;num);     // Finds last digit of the number and print it in words     while(num!=0)     {         switch(num%10) {             case 0: printf("Zero "); break;             case 1: printf("One "); break;</pre>	<pre>            case 2: printf("Two "); break;             case 3: printf("Three "); break;             case 4: printf("Four "); break;             case 5: printf("Five "); break;             case 6: printf("Six "); break;             case 7: printf("Seven "); break;             case 8: printf("Eight "); break;             case 9: printf("Nine ");         }         //end of switch         num = num/10;     }     //end of while loop }</pre>
--	--

#### Exercise Problems:

1. Write a C program to find Greatest Common Divisor (GCD ) of two given integers. GCD of two integers is the highest number that totally divides those two integers. E.g. GCD of 15 and 25 is 5.
2. Write a C program to compute the quadruple factorial of a given number  $n$ ,  $q(n) = \frac{(2n)!}{n!}$
3. Write a C program to compute the super-factorial of a given number  $n$ ,  $sf(n) = 1! * 2! * 3! * ... n!$

#### Assignment Problems:

1. Write a C program to display a given number in words starting from its leftmost digit.  
**Hint:** Compute the reverse of the given number and then use a while loop like practice 3 to print the digits.  
E.g., if input number is 1234 your program should print "One Two Three Four".
2. Write a C program to convert a given Binary number to its Decimal equivalent.
3. Write a C program to convert a given decimal number to its binary equivalent.
4. Write a C program to compute the sum of the series:  $1/1! + 1/2! + 1/3! + ... + 1/n!$  where n is an input.
5. Write a C program that prints all even numbers between m and n (m,n are user inputs) except the ones which are divisible by 3. [*Hint:* Use continue statement within a loop] Sample input/output:  
Enter m: 99  
Enter n: 116  
All even numbers between 100 and 112 except those divisible by 3 are: 100, 104, 106, 110, 112, 116,
6. Write a C program that asks a shopper to enter amount (in kg) and total price of sugar he bought from different places. If the shopper mistakenly enters a negative number as amount/price, it prints an error message "Invalid input, enter a positive number" and prompts the shopper to give another input. When the shopper enters 0 as an amount then the program terminates and shows the shopper total amount, price and average price of sugar per kg. [*Hint:* Use continue statement within a loop] Sample input/output:  
Enter amount (in kg): 5  
Enter price: 350  
Enter amount (in kg): -3  
*Invalid input, enter a positive number*  
Enter amount (in kg): 5  
Enter price: -67  
*Invalid input, enter a positive number*  
Enter amount (in kg): 10  
Enter price: 650  
Enter amount (in kg): 0  
Total amount (in kg): 15, Total price: 1000, Average price per kg: 66.67