

Indian Institute of Technology Bombay, Mumbai
Department of CSE, Kanwal Rekhi Building
CS101 – Computer Programming
Autumn Semester 2014-2015

Lab Handout for Week 4 - 18/08/2014 to 22/08/2014

Objective: In this lab, you will practise writing, modifying, compiling, and executing C++ programs. The programs deal with Loops. You required to read, solve, compile and execute the programs mentioned in this lab handout.

Programming Examples

1. You want to buy a laptop worth Rs.30000/-. But your current savings are only Rs.5000/-. Given that the cost price of laptop increases by 2% every month and you can increase your savings by 7% monthly, how many months would you need to wait to buy that laptop. Try the problem for user given monthly increases as well. Identify first which looping construct suits your needs best.

```
#include <iostream>
using namespace std;
int main( void ) {
    float priceOfLaptop = 30;
    float savings = 5;
    int months = 0;
    while (priceOfLaptop > savings) {
        priceOfLaptop += (2 * priceOfLaptop) / 100;
        savings += (7 * savings) / 100;

        ++months;
    }
    cout << "Price of Laptop : " << priceOfLaptop << endl;
    cout << "Savings : " << savings << endl;
    cout << "Months : " << months;
    return 0;
}
```

Filename: *buy_laptop.cpp*

2. Create a program that takes as input the length and breadth of a rectangle. And outputs a rectangle using a asterisk(*).

Suppose if you input 7 and 4 the output should be like

* *

* *

```
#include<iostream>
using namespace std;
int main() {
    int length,height,i,j;
    cout<<"Enter length and height:";
    cin>>length>>height;
    for(i=0;i<height;i++) {
```

```

        if(i==0 || i==height-1) {
            for (j=0;j<length;j++)
                cout<<"*";
        }
        else {
            for(j=0;j<length;j++) {
                if(j==0 || j==length-1) {
                    cout<<"*";
                }
                else {
                    cout<<" ";
                }
            }
        }
        cout<<endl;
    }
    return 0;
}

```

Filename: draw_rectangle.cpp

3. Write a program to find the second highest number from `n` positive numbers given in input.

```

#include<iostream>
using namespace std;
int main(){
    int i, n, max, max2;
    cout<<"Please enter n: ";
    cin>>n;
    max = 0;
    max2 = 0;
    for(i = 0; i < n; i++) {
        int temp;
        cin >> temp;
        if (max < temp) {
            max2 = max;
            max = temp;
        }
        else if (max2 < temp) {
            max2 = temp;
        }
    }
    cout<<"The second highest was: "<<max2<<endl;
}

```

Filename: second_highest.cpp

4. Write a program for conversion of Decimal to Binary number. Conversion of decimal to binary number is based on collecting remainders and iterative division by 2

sample input-output:

Enter the decimal to be converted:16

The binary of the given number is:10000

```
#include <iostream>
using namespace std;
int main(){
    long dec,rem,i=1,sum=0;
    cout<<"Enter the decimal to be converted:";
    cin>>dec;
    do{
        rem=dec%2;
        sum=sum + (i*rem);
        dec=dec/2;
        i=i*10;
    }while(dec>0);
    cout<<"The binary of the given number is:"<<sum<<endl;
    return 0;
}
```

Filename: second_highest.cpp

Programming Exercises

1. Predict the output for this code without executing it.

```
#include<iostream>
using namespace std;
int main(){
    for(int i=0;i<10;i++){
        cout<<i<<endl;
    }
    for(int i=0;i<10;){
        cout<<i<<endl;
    }
    for(int i=0;;i++){
        cout<<i<<endl;
    }
    for(;;){
        cout<<"Hello"<<endl;
    }
}
```

Filename: for_loop_without_execute.cpp

Solution:

0123456789

000000000.....(Infinite times)

012345678.....(Infinite times)

HelloHelloHello.....(Infinite times)

2. The following program calculates a factorial. Replace the for loop with a while loop.

```
#include<iostream>
using namespace std;
int main(){
    int n, i;
    cout<<"Please enter n: ";
    cin>>n;
    int factorial = 1;
    for(i = 1; i <= n; i++) {
        factorial *= i;
    }
    cout<<n<<"! is "<<factorial<<endl;
}
```

Filename: *factorial_using_for.cpp*

Solution:

```
#include<iostream>
using namespace std;
int main() {
    int n, i;
    cout<<"Please enter n: ";
    cin>>n;
    int factorial = 1;
    i = 1;
    while (i <= n) {
        factorial *= i;
        i++;
    }
    cout<<n<<"! is "<<factorial<<endl;
}
```

Filename: *factorial_using_while.cpp*

3. Print the following pattern using loops. [Refer example 2 for help]

```
*
***
*****
*****
```

Solution:

```
#include <iostream>
using namespace std;
int main(){
    for(int i=0;i<4;i++){
        for(int j=i;4-j-1 > 0;j++)
            cout << " ";
        for(int k=0;k<2*i+1;k++)
            cout << "*" ;
        cout << endl;
    }
}
```

Filename: *pattern.cpp*

4. Write a C++ program to check whether a number entered by the user is an Armstrong number.

[Armstrong Number is a number for which sum of cubes of its digits is equal to the number itself. Eg. $371 = 3^3 + 7^3 + 1^3$]

Solution:

```
#include <iostream>
using namespace std;
int main(){
    int num,rem,temp,temp1=0;
    cin >> num;
    temp=num;
    while(temp>0){
        rem=temp%10;
        temp1=temp1+rem*rem*rem;
        temp=temp/10;
    }
    if (num==temp1) cout << "Armstrong number";
    else cout << "Not an Armstrong number"
}
```

Filename: armstrong.cpp

Additional Exercises(optional)

1. Write a C++ program to take a natural number, n as input from the user. Check whether the number is a palindrome or not and print the result.

A number is a palindrome if it reads the same in forward and backward direction.

Example.

12321 is a palindrome.

565 is a palindrome.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    int i,a,n=0;
    cout<<"\nEnter a natural number : ";
    cin>>i;
    a = i;
    while(a > 0) {
        n *= 10;
        n += a % 10;
        a /= 10;
    }
    cout<<"\nNumber is ";
    if(n != i)
        cout<<"not ";
    cout<<"a palindrome";
    return 0;
}
```

Filename: palindrome.cpp

2. Write a C++ program to take a natural number, n as input from the user. Now print first n prime numbers.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    int n,i=0,j=2,k,flag;
    cout<<"\nEnter a natural number : ";
    cin>>n;
    while( i < n ) {
        for( ; ++j ) {
            flag = 0;
            for( k=2; k <= j / 2; ++k ) {
                if( j % k == 0 ) {
                    flag = 1;
                    break;
                }
            }
            if( flag == 0 ) {
                cout<<endl<< j;
                break;
            }
        }
        j = j + 1;      ++i;
    }
    return 0;
}
```

Filename: *prime_number.cpp*

3. Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power() that takes a double value for n and an int value for p, and returns the result as a double value. Use a default argument of 2 for p, so that if this argument is omitted, the number n will be squared. Write a main() function that gets values from the user to test this function. Hint: Declare the function so that it will take value of p as 2.

Solution:

```
#include<iostream>
#include<string>
#include<stdlib.h>
#include<sstream>
using namespace std;
double power(double n, int p=2);
int main(){
    int n; int p=2; string s;
    cout<<"Enter n: ";
    cout<<"Enter p: ";
    int i=0;
    getline(cin,s);
    istringstream iss(s);
    while(iss) {
        string sub;
        iss >> sub;
        if(i==0){
            stringstream(sub) >> n;      }
    }
```

```

        else{
            stringstream(sub) >> p;
        }
        i++;
    }
    cout <<"The power is "<<power(n, p)<<endl;
    return 0;
}
double power(double n, int p){
    int ret;
    for(ret=1; p>0; p--)
        ret*=n;
    return ret;
}

```

Filename: power.cpp

4. Calculate nCr . Take inputs from user and check for base conditions like $r \leq n$ and $r \geq 0$.

Solution:

```

#include<iostream>
using namespace std;
int main(){
    int n,r,numerator1=1,denominator1=1,denominator2=1;
    cout<<"enter n:"<<endl;
    cin>>n;
    cout<<"enter r:"<<endl;
    cin>>r;
    if(r > n){
        cout<<"wrong input! value of r should be less than or equal to n"<<endl;
        return -1;
    }
    else if(r < 0){
        cout<<"wrong input! value of r should be greater than or equal to 0"<<endl;
        return -1;
    }
    else if(r == 0){
        cout<<"nCr=1"<<endl;
        return 0;
    }
    for(int i=1;i<=n;i++){
        numerator1=numerator1*i;
    }
    for(int j=1;j<=r;j++){
        denominator1=denominator1*j;
    }
    for(int k=1;k<=n-r;k++){
        denominator2=denominator2*k;
    }
    int comb=numerator1/(denominator1*denominator2);
    cout<<"nCr="<<comb<<endl;
    return 0;
}

```

Filename: ncr.cpp