



INDIAN ASSOCIATION FOR THE CULTIVATION OF SCIENCE
SCHOOL OF MATHEMATICAL & COMPUTATIONAL SCIENCES

PG - I (SMCS), Autumn Semester 2022-23

Instructor: **Debarshi Kumar Sanyal**

Assignment Number: **2**

Course: **COM 4111: Object-Oriented Programming with C++**

Date: **Sep 10, 2022**

INSTRUCTIONS

- (1) Create a folder with the name **Assignment_1_\$name\$** (where **\$name\$** denotes name of the student). In that folder create sub-folders for each problem. (example: **prob_0**, **prob_1**, **prob_2**, etc.). Write your codes for a particular problem in the respective folder (e.g., **prob1** might contain two C++ programming files **swap.cpp** and **swap2.cpp**)
 - (2) Give meaningful comments to explain the method used in your program. Programs without valid comments will be evaluated to zero marks.
-

Problem 1 Explain the compilation errors / output of each program given below.

(i)

```
#include <iostream >
using namespace std;
int main () {
    int a;
    a = -5;
    char a = 'A';
    cout << a << "\n";
    return 0;
}
```

- (ii)

```
include <iostream >
using namespace std;
int main () {
    int a;
    a = -1;
    {
        char a = 'A';
        cout << arg1 << "\n";
    }
    return 0;
}
```
- (iii)

```
include <iostream >
using namespace std;
int main () {
    int a;
    a = -1;
    {
        char a = 'A';
    }
    cout << arg1 << "\n";
    return 0;
}
```
- (iv)

```
void f(double val, int val2) {
    int x = val;
    int x2 {val};
    char c {val2};
    char c2 {24};
    char c3 {264};
    int x3 {2.0};
    int x4 { };
    int x4[3] { };
}
int main() {
    double d = 5.0;
    int i = 2;
    f(d,i);
    return 0;
}
```
- (v)

```
void f(char* p) {
    char s[] = "Gorm";

    const char* pc = s;
    pc[3] = 'g';
}
```

```

pc=p;

char *const cp = s;
cp[3] = 'a';
cp = p;

const char *const cpc = s;
cpc[3] = 'a';
cpc = p;

const int a = 5;
const int* pa1 = &a;
int* pa2 = &a;

}
int main() {
    char a[] = "IACS";
    f(a);
    return 0;
}

```

```

(vi) void f() {
    int var = 1;
    int& r {var};
    int x = r;
    r = 2;
    int& r2;
    ++r;
    int* pp = &r;
    extern int& r3;
    int& ei = 99;
    const int& ei2 = 99;
}
int main() {
    f();
    return 0;
}

```

Problem 2

1. Write a C++ program to calculate the factorial of an input integer. Store the input and the factorial in an `int`. [factorialPlain.cpp]
2. What is the largest input for which your program works?

3. Suppose your input is an integer between 0 and 8. You do not want to execute the expensive task of multiplications when the user enters an integer as input. Can you re-implement your program to achieve this? [factorialQuick.cpp]

Problem 3

1. Write a C++ function (and program) to copy an array **A** to another array **B**, both of **N** elements. [copyArray.cpp]
2. Write a C++ function (and program) to reverse an array **A** of **N** elements. [reverseArray.cpp]

Problem 4 Read about bitwise operators in C++. Write programs to achieve the following for unsigned integers **a** and **b**. [BitOp.cpp]

1. Multiply **a** by 8 using shift operations.
2. Count the number of 1's in the binary representation of **a**.
3. Count the number of trailing 0's in the binary representation of **a**.
4. Reverse the number **a** by reversing the bits of **a**.
5. Print the bitwise AND, OR and XOR of **a** and **b**.

Problem 5 You are given an array of elements, where each element is an unsigned integer. All elements occur an even number of times except one number, say, **x**. Write a program in C++ using bitwise operators to find **x**. [oddNumB0.c]

Problem 6 Read about little-endian and big-endian formats. You are given an unsigned integer initialized to 1:

```
#include <iostream>
using namespace std;
int main() {
    unsigned int i = 1;
    //...
}
```

Complete the above C++ program to determine if your system follows little-endian or big-endian format. [endianTest.cpp]