	COMSATS University Islamabad, Vehari Campus Department of Computer Science

Class: BCS-SP22-4B **Submission Deadline: 10 Sep 2023**

Subject: Data Structures and Algorithms-Lab **Instructor: Yasmeen Jana**

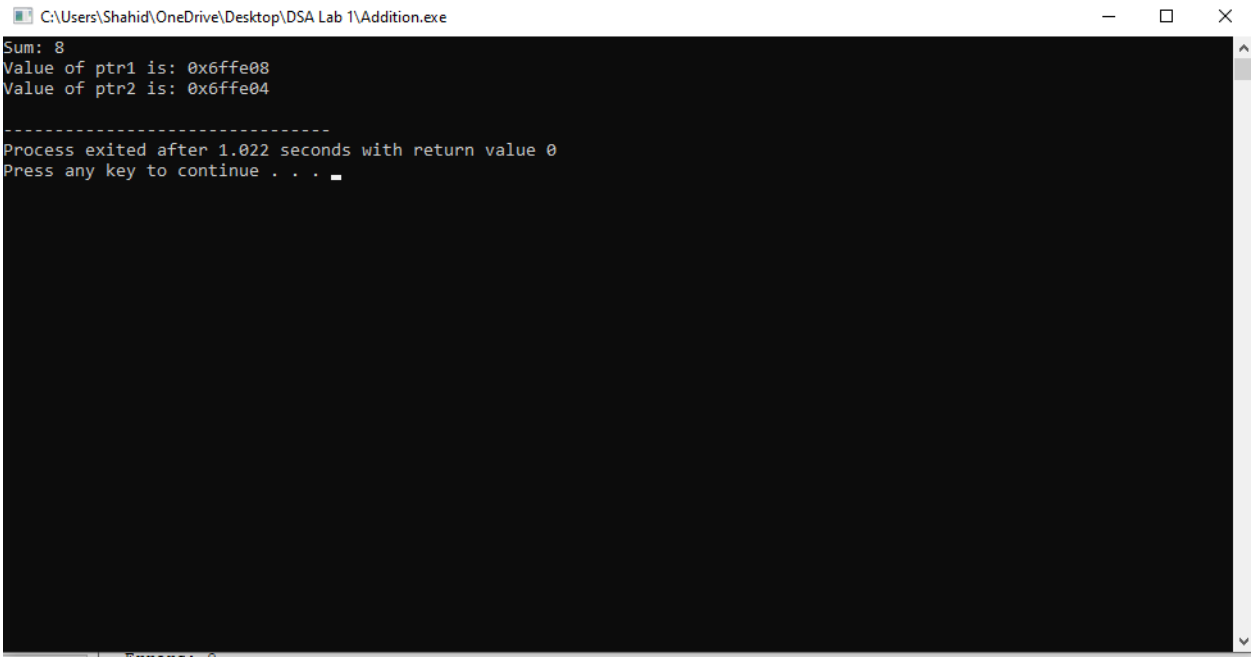
Max Marks: 10 **Reg. No: SP22-BCS-017**

GitHub Link:

https://github.com/shahid-Azlan/DSA-LAB-1

Problem 1:

```
#include <iostream>
using namespace std;
int main() {
int a = 5, b = 3;
int *ptr1 = &a, *ptr2 = &b;
int result = *ptr1 + *ptr2;
cout << "Sum: " << result <<endl;
cout << "Value of ptr1 is: "<<ptr1<<endl;
cout << "Value of ptr2 is: "<<ptr2<<endl;
return 0;
}
```

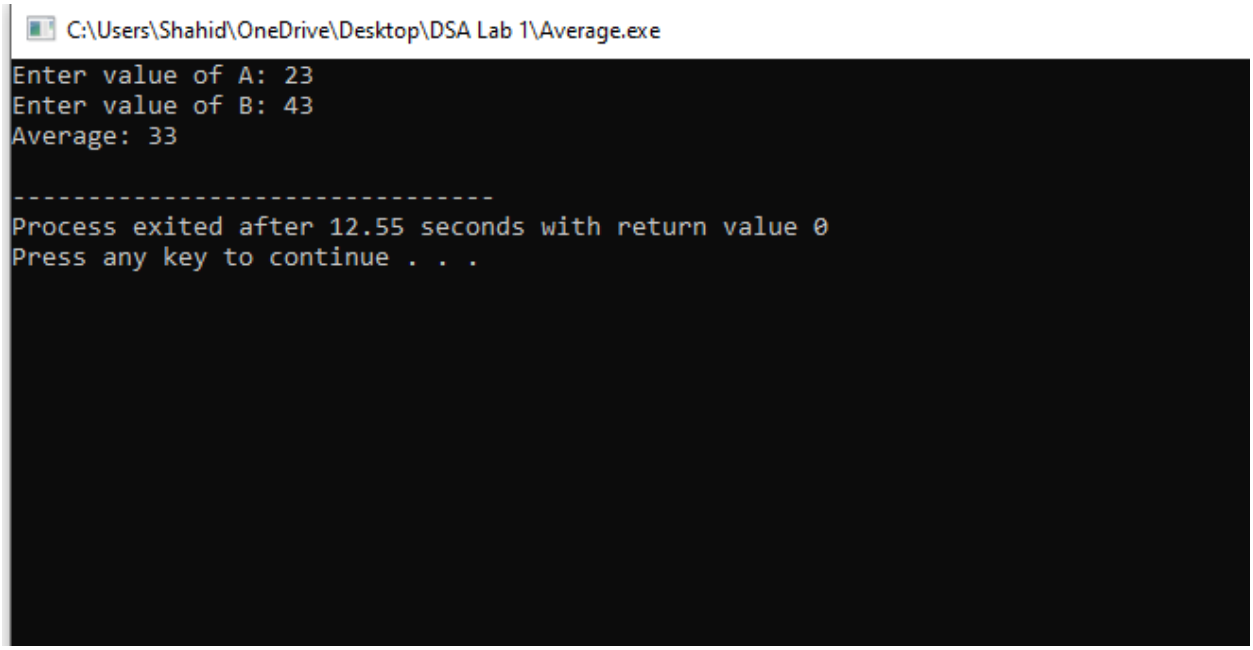


Problem 2:

```
#include <iostream>
using namespace std;
int main() {
int a;
int b;
cout<<"Enter value of A: ";
cin >> a;
cout<<"Enter value of B: ";
cin >> b;
int *arr1 = &a;
int *arr2 = &b;
int sum = *arr1 + *arr2;
float average=sum/2.0;
cout << "Average: " << average << endl;
```

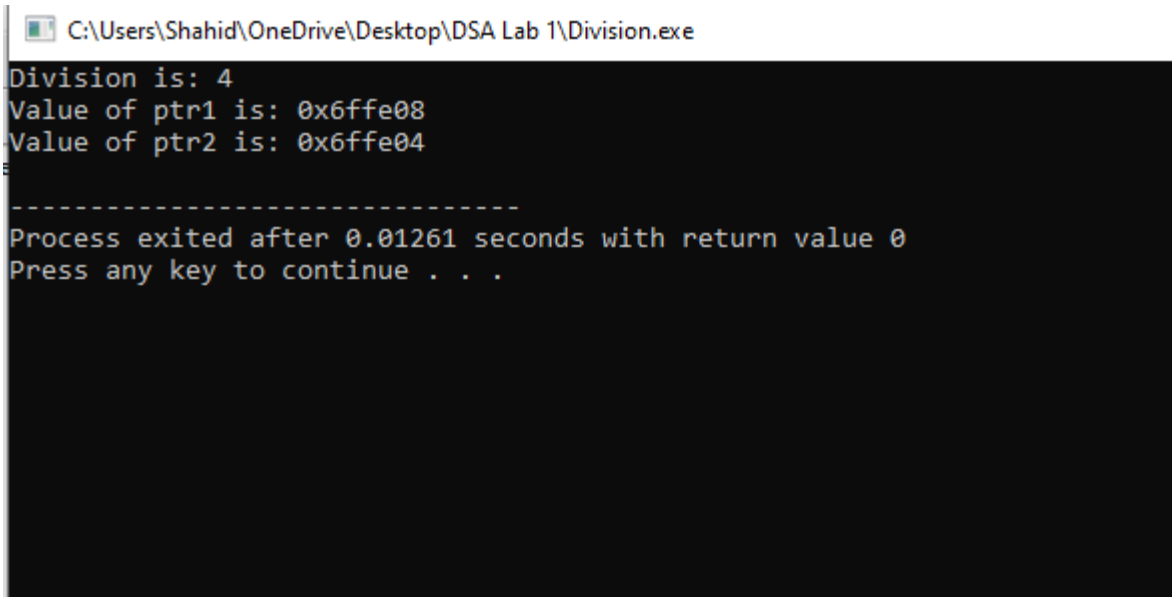
return 0;

}



Problem 3:

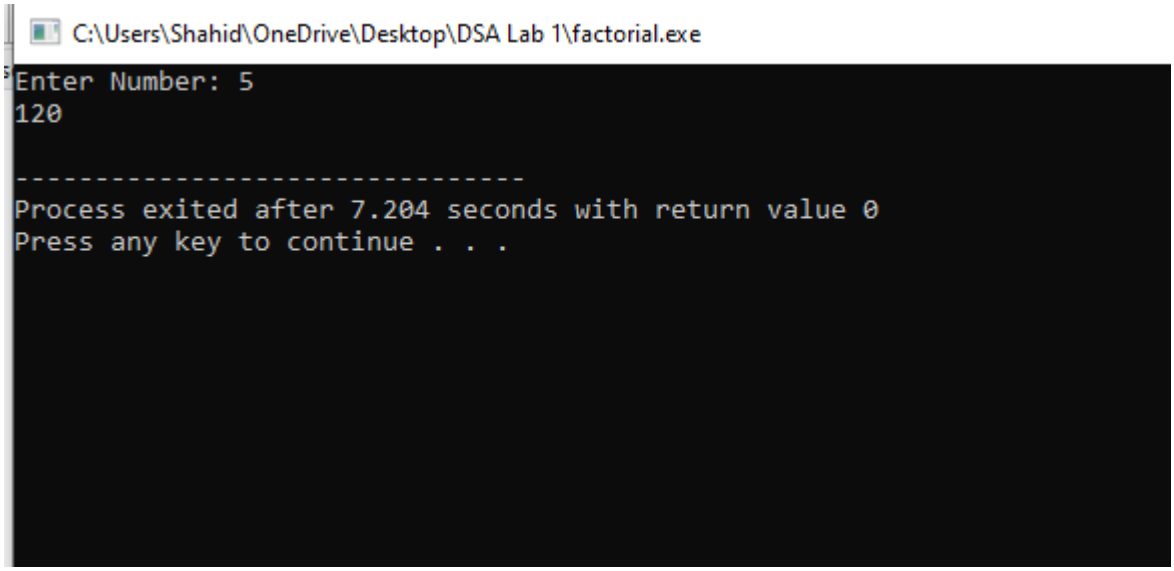
```
#include <iostream>
using namespace std;
int main() {
int a = 28, b = 7;
int *ptr1 = &a, *ptr2 = &b;
int result = *ptr1 / *ptr2;
cout << "Division is: " << result << endl;
cout << "Value of ptr1 is: "<<ptr1<<endl;
cout << "Value of ptr2 is: "<<ptr2<<endl;
return 0;
}
```



Problem 4:

```
#include <iostream>
using namespace std;
int factorial(int* n) {
int result = 1;
for (int i = 1; i <= *n; i++) {
result *= i;
}
return result;
}
int main() {
int num;
cout<<"Enter Number: ";
cin >> num;
int* ptr_num = &num;
```

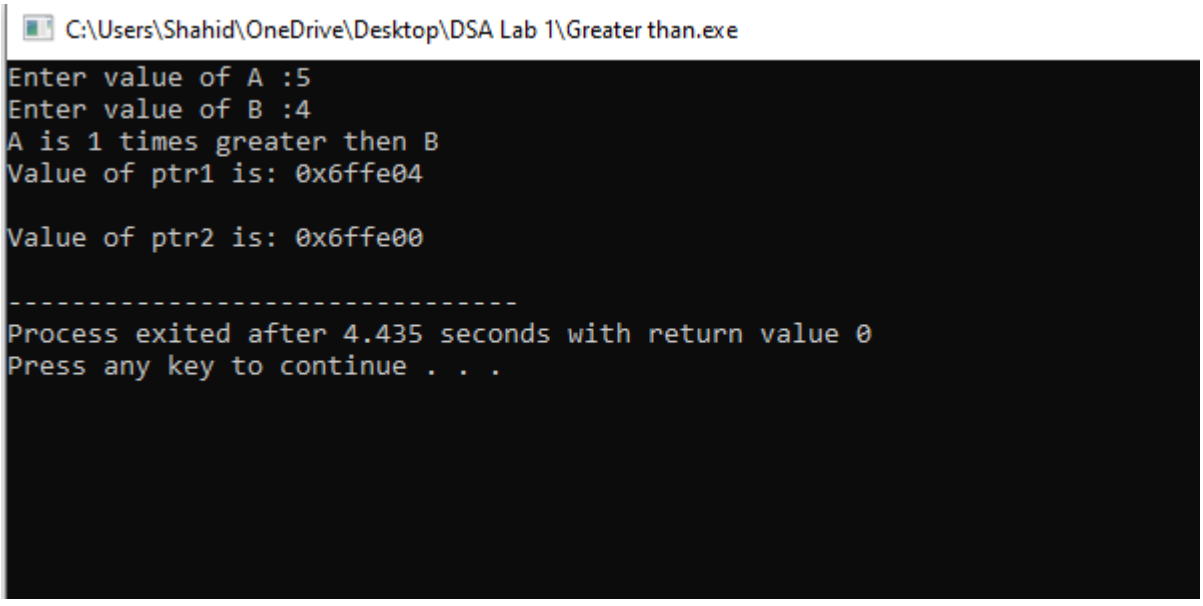
```
int result = factorial(ptr_num);
cout << result << endl;
return 0;
}
```



Problem 5:

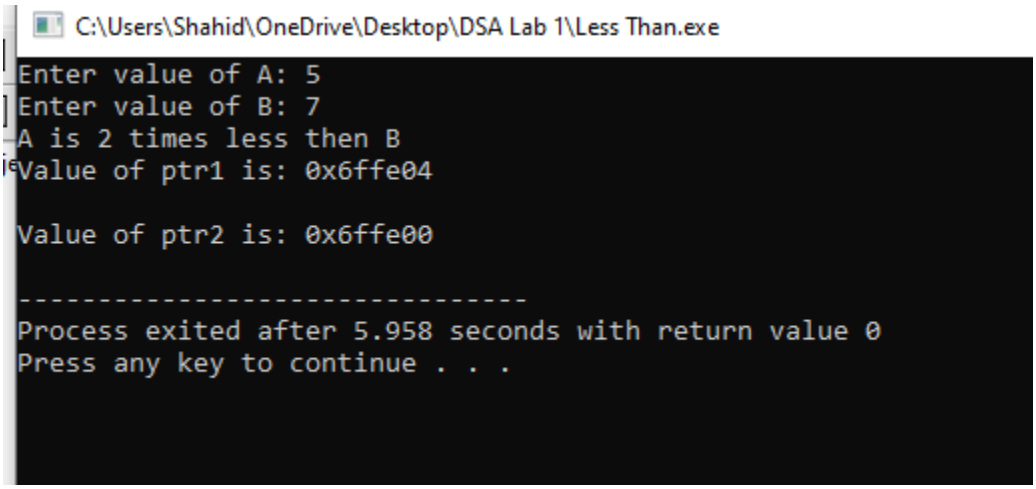
```
#include <iostream>
using namespace std;
int main() {
int a;
int b;
cout<<"Enter value of A :";
cin>>a;
cout<<"Enter value of B :";
cin>>b;
int *ptr1 = &a, *ptr2 = &b;
if (*ptr1==*ptr2){
```

```
cout<<"Both numbers are equal";
} else if (*ptr2>*ptr1){
int d=*ptr2-*ptr1;
cout<<"B is "<<d<<" greater than A";
}
else{
int c=*ptr1-*ptr2;
cout<<"A is "<<c<<" times greater then B";
}
cout << "\nValue of ptr1 is: "<<ptr1<<endl;
cout << "\nValue of ptr2 is: "<<ptr2<<endl;
return 0;
}
```



Problem 6:

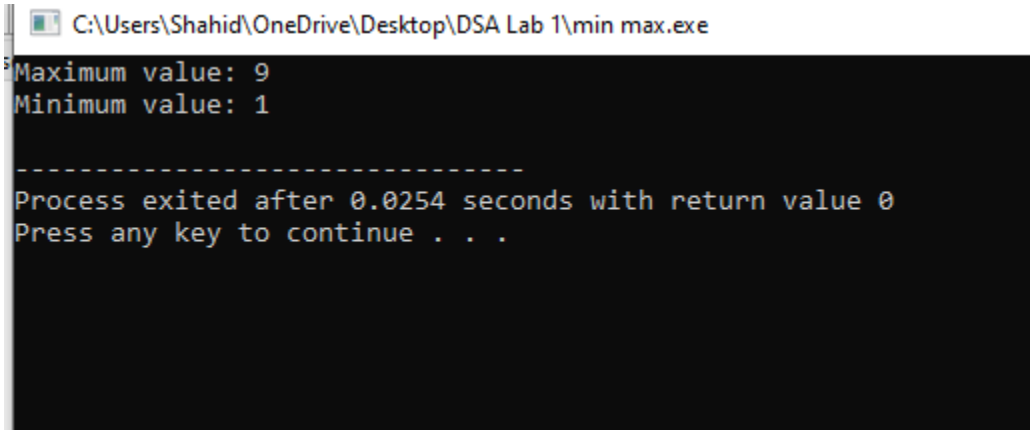
```
#include <iostream>
using namespace std;
int main() {
int a;
int b;
cout<<"Enter value of A: ";
cin>>a;
cout<<"Enter value of B: ";
cin>>b;
int *ptr1 = &a, *ptr2 = &b;
if (*ptr1==*ptr2){
cout<<"Both numbers are equal";
} else if (*ptr2<*ptr1){
int d=*ptr1-*ptr2;
cout<<"B is "<<d<<" times less than A";
}
else{
int c=*ptr2-*ptr1;
cout<<"A is "<<c<<" times less then B";
}
cout << "\nValue of ptr1 is: "<<ptr1<<endl;
cout << "\nValue of ptr2 is: "<<ptr2<<endl;
return 0;
}
```



Problem 7:

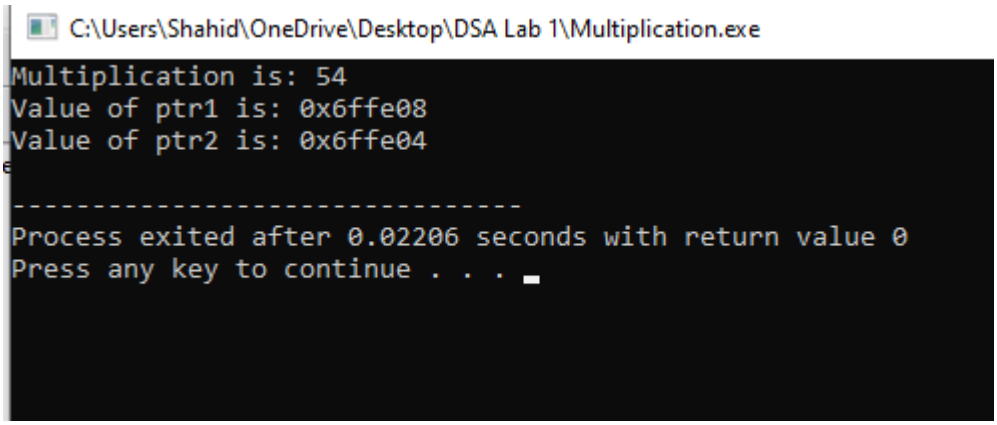
```
#include <iostream>
using namespace std;
int main() {
int arr[] = {5, 2, 9, 1, 7, 3, 8, 4, 6};
int n = sizeof(arr) / sizeof(arr[0]);
int *ptr = arr;
int max_val = *ptr;
int min_val = *ptr;

for (int i = 1; i < n; i++) {
if (*(ptr + i) > max_val) {
max_val = *(ptr + i);
}
if (*(ptr + i) < min_val) {
min_val = *(ptr + i);
}
}
```

```
cout << "Maximum value: " << max_val << endl;
cout << "Minimum value: " << min_val << endl;

return 0;
}
```



Problem 8:

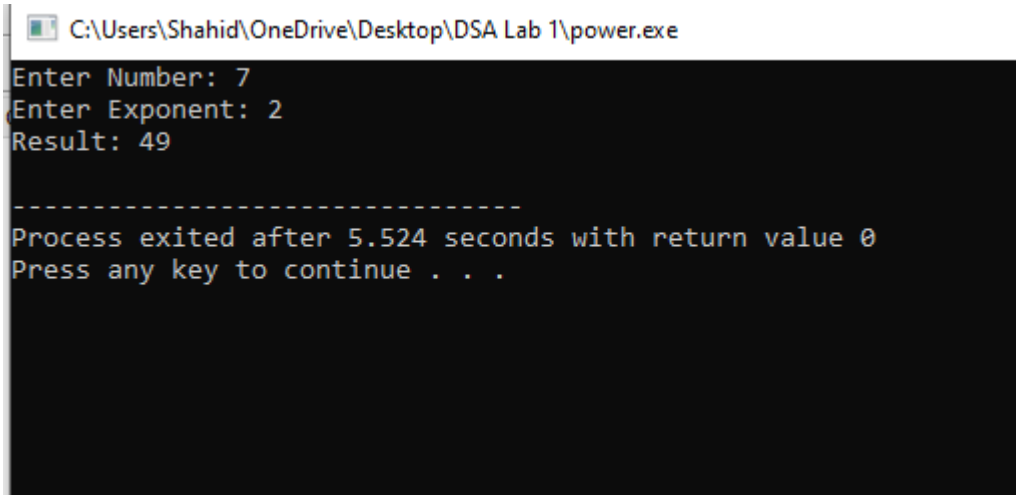
```
#include <iostream>
using namespace std;
int main() {
int a = 6, b = 9;
int *ptr1 = &a, *ptr2 = &b;
int result = *ptr1 * *ptr2;
cout << "Multiplication is: " << result << endl;
cout << "Value of ptr1 is: "<<ptr1<<endl;
cout << "Value of ptr2 is: "<<ptr2<<endl;
return 0;
}
```



Problem 9:

```
#include <iostream>
using namespace std;
double power(double *base, int *exponent) {
double result = 1.0;
int exp = *exponent;

if (exp >= 0) {
for (int i = 0; i < exp; i++) {
result *= *base;
}
} else {
for (int i = 0; i < -exp; i++) {
result /= *base;
}
}

return result;
}
```

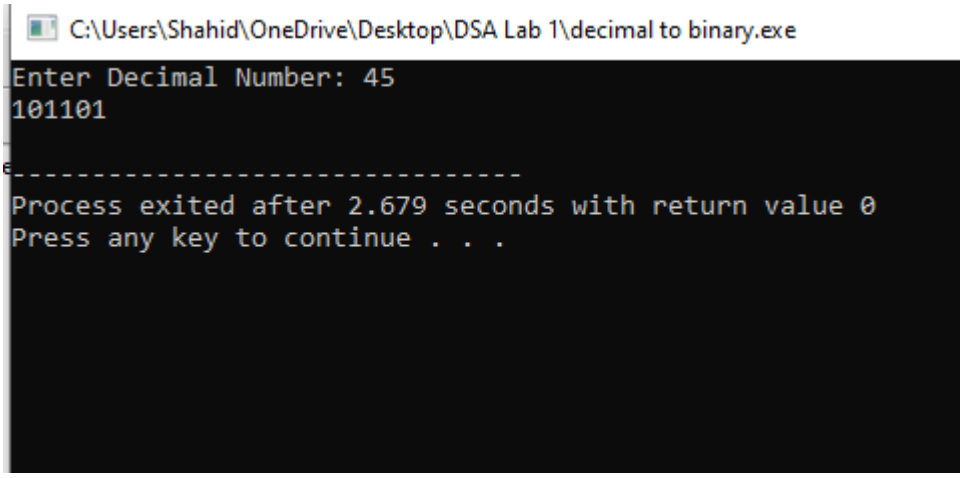
```
int main() {
 double base;
 int exponent;
        cout<<"Enter Number: ";
 cin >> base;
        cout<<"Enter Exponent: ";
            cin >> exponent;
    double *ptr_base = &base;
  int *ptr_exponent = &exponent;

double result = power(ptr_base, ptr_exponent);

cout << "Result: " << result << endl;

return 0;
}
```



Problem 10:

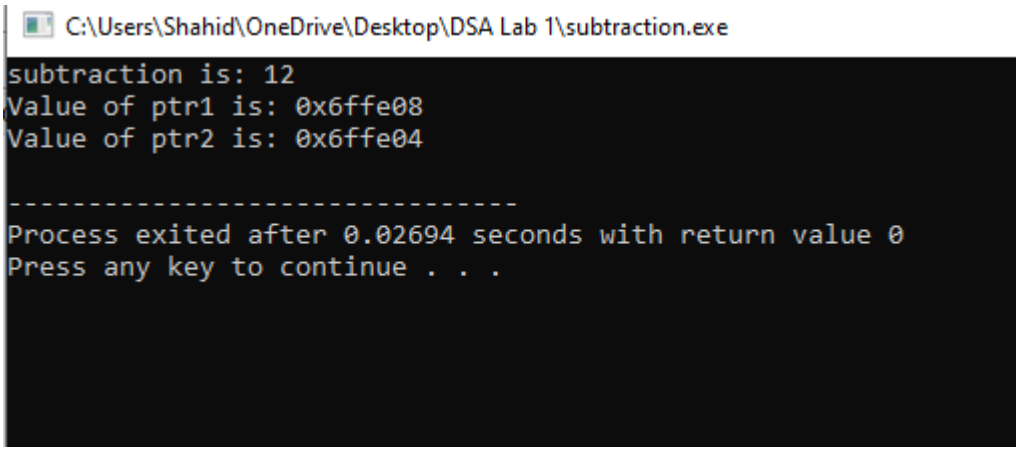
```
#include <iostream>
using namespace std;
void decimalToBinary(int n) {
int binary[32];
int* ptr = binary;

int i = 0;
while (n > 0) {
*(ptr + i) = n % 2;
n /= 2;
i++;
}

for (int j = i - 1; j >= 0; j--) {
cout << *(ptr + j);
}
}

int main() {
int decimal;
cout<<"Enter Decimal Number: ";
cin >> decimal;

decimalToBinary(decimal);
```

```
cout << endl;


return 0;

}
```



Problem 11:

```
#include <iostream>

using namespace std;

int main() {

int a = 24, b = 12;

int *ptr1 = &a, *ptr2 = &b;

int result = *ptr1 - *ptr2;

cout << "subtraction is: " << result << endl;

cout << "Value of ptr1 is: "<<ptr1<<endl;

cout << "Value of ptr2 is: "<<ptr2<<endl;

return 0;

}
```

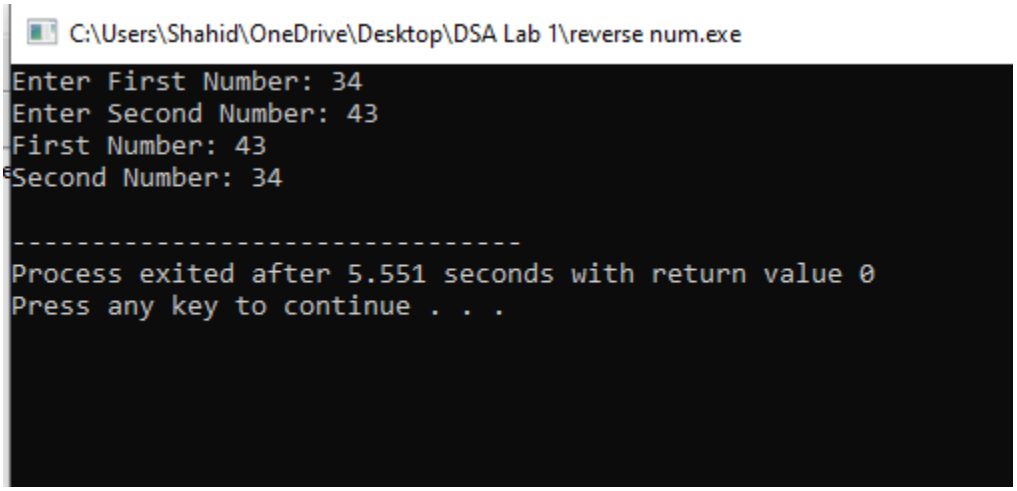


Problem 12:

```
#include <iostream>
using namespace std;
int main() {
int a, b;
cout <<"Enter First Number: ";
cin >> a;
cout <<"Enter Second Number: ";
cin >> b;
int* ptrA = &a;
int* ptrB = &b;

*ptrA = *ptrA + *ptrB;
*ptrB = *ptrA - *ptrB;
*ptrA = *ptrA - *ptrB;

cout << "First Number: "<<*ptrA <<"\nSecond Number: "<<*ptrB << endl;
```

return 0;

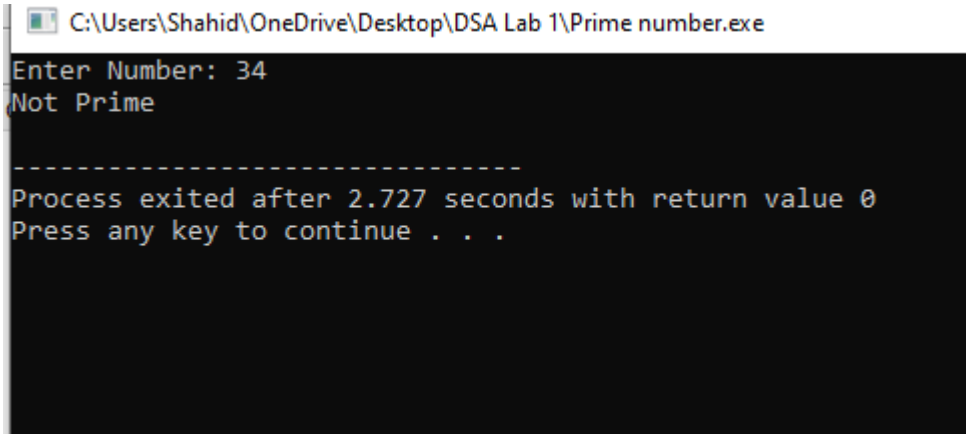
}



Problem 13:

```
#include <iostream>
using namespace std;
bool isPrime(int *n) {
if (*n <= 1) {
return false;
}

for (int i = 2; i <= (*n) / 2; i++) {
if ((*n) % i == 0) {
return false;
}
}

return true;
}
```

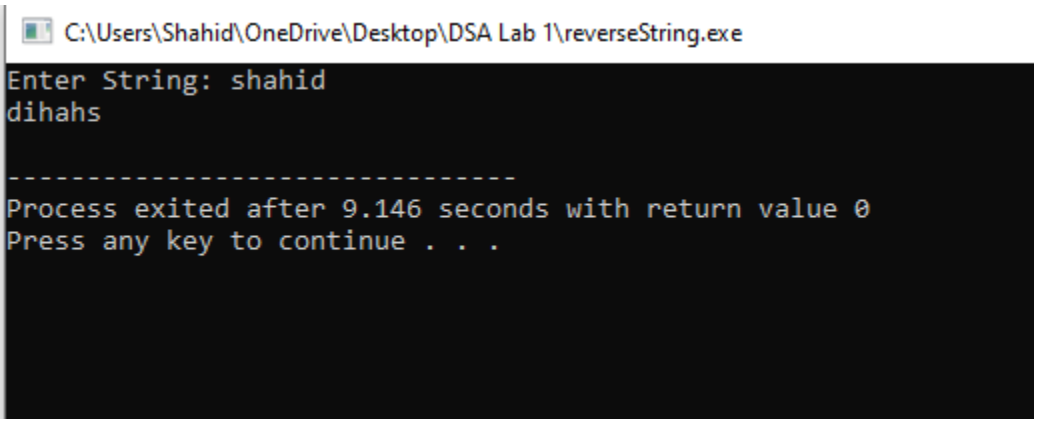
```
int main() {
int num;
cout<<"Enter Number: ";
cin >> num;
int *ptr_num = &num;

if (isPrime(ptr_num)) {
cout << "Prime" << endl;
} else {
cout << "Not Prime" << endl;
}

return 0;
}
```



Problem 14:

```
#include <iostream>
using namespace std;
void reverseString(char *str) {
```

```
char *start = str;
char *end = str;

while (*end != '\0') {
end++;
}
end--;

while (start < end) {
char temp = *start;
*start = *end;
*end = temp;
start++;
end--;
}
}

int main() {
char str[100];
cout<<"Enter String: ";
cin >> str;
reverseString(str);
cout << str << endl;
return 0;
}
```



Problem 15:

```
#include <iostream>
using namespace std;
int main() {
  int x = 10;
int* ptr = &x;
*ptr += 5;
cout <<"Value of x is: "<<x<< endl;
cout <<"Value of increment x is: "<< *ptr << endl;
cout <<"address of ptr is: "<< ptr << endl;
return 0;
}
```