Ahsania Mission University of Science & Technology

Lab Report

Lab No: 02

Course Code: CSE 2202

Course Title: Computer Algorithm Sessional.

Submitted By:

MT.Sumaiya Ajmeri Borsha
ID: 1012320005101020
1st Batch, 2nd Year, 2nd Semester
Department of Computer science and Engineering,
Ahsania Mission University of Science & Technology

Submitted To:

Md. Fahim Faisal
Lecturer,
Department of Computer science and Engineering,
Ahsania Mission University of Science & Technology

Task No.: 01

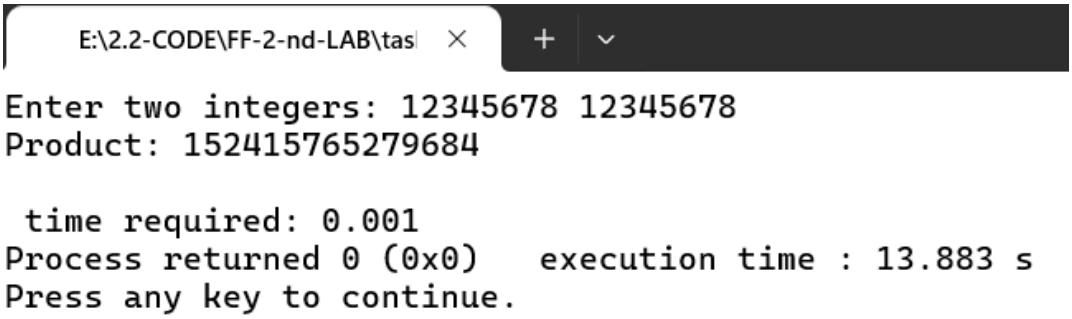
Problem Statement: Integer Multiplication and Use the time reading function for time calculation.

Source Code:

```
#include <iostream>
#include <cstring>
using namespace std;
#define MAX 200
#include<ctime>
class BigIntMultiplication
{
private:
   int numA[MAX], numB[MAX], result[MAX];
   int lenA, lenB;
public:
   BigIntMultiplication()
   {
      memset(numA, 0, sizeof(numA));
      memset(numB, 0, sizeof(numB));
      memset(result, 0, sizeof(result));
      lenA = lenB = 0;
   }
   void storeNumber(int num, int arr[], int &length)
   {
      while (num > 0)
      {
         arr[length++] = num % 10;
         num /= 10;
      }
   }
   void multiply(int A, int B)
   {
      if (A == 0 || B == 0)
      {
         cout << "0" << endl;
         return;
      }
      storeNumber(A, numA, lenA);
```

```
        storeNumber(B, numB, lenB);
        for (int i = 0; i < lenA; i++)
        {
            for (int j = 0; j < lenB; j++)
            {
                result[i + j] += numA[i] * numB[j];
                result[i + j + 1] += result[i + j] / 10;
                result[i + j] %= 10;
            }
        }
        printResult();
    }

    void printResult()
    {
        int lenResult = lenA + lenB;
        while (lenResult > 1 && result[lenResult - 1] == 0)
        {
            lenResult--;
        }
        for (int i = lenResult - 1; i >= 0; i--)
        {
            cout << result[i];
        }
        cout << endl;
    }
};

int main()
{
    int A, B;
    cout << "Enter two integers: ";
    cin >> A >> B;
    BigIntMultiplication multiplier;
    cout << "Product: ";
    clock_t time1 = clock();
    multiplier.multiply(A, B);
    clock_t time2 = clock();
    double time_spand = double(double(time2-time1)/ CLOCKS_PER_SEC);
    cout<<"\n time required: "<<time_spand;
    return 0;}
```

Output:



Task No.: 02

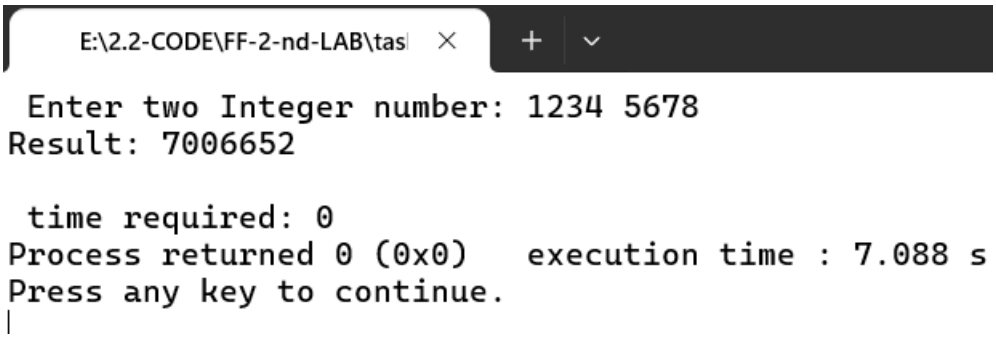
Problem Statement: Using Karatsuba Multiplication algorithm write a program that multiply two Integer number.

Source Code:

```
#include <iostream>
#include <cmath>
#include<ctime>
using namespace std;

int power(int x, int n) {
   return pow(10, n);
}

int karatsuba(int x, int y) {

   if (x < 10 || y < 10) {
      return x * y;
   }

   int n = max(log10(x) + 1, log10(y) + 1);
   int half = n / 2;

   int a = x / power(10, half);
   int b = x % power(10, half);
```

```
    int c = y / power(10, half);
    int d = y % power(10, half);
    int ac = karatsuba(a, c);
    int bd = karatsuba(b, d);
    int ad_plus_bc = karatsuba(a + b, c + d) - ac - bd;
    return ac * power(10, 2 * half) + ad_plus_bc * power(10, half) + bd;
}

int main() {
    int num1, num2;
    cout<<" Enter two Integer number: ";
    cin>>num1>>num2;
    clock_t time1 = clock();
    int result = karatsuba(num1, num2);
    clock_t time2 = clock();
    cout << "Result: " << result << endl;
    double time_spand = double(double(time2-time1)/ CLOCKS_PER_SEC);
    cout<<"\n time required: "<<time_spand;
    return 0;
}
```

Output:



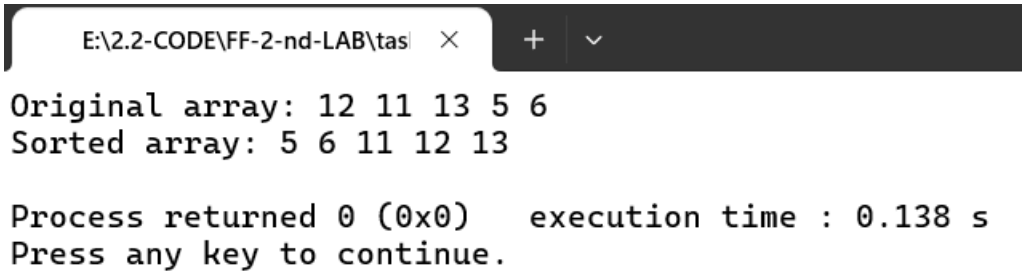
Task No.: 03

Problem Statement: Implement the insertion sort algorithm for sorting one dimensional array.

Source Code:

```
#include <iostream>
using namespace std;

void insertionSort(int arr[], int n) {

   for (int i = 1; i < n; i++) {
      int key = arr[i];
      int j = i - 1;
      while (j >= 0 && arr[j] > key) {
         arr[j + 1] = arr[j];
         j = j - 1;
      }
      arr[j + 1] = key;
   }
}
void printArray(int arr[], int n) {
   for (int i = 0; i < n; i++) {
      cout << arr[i] << " ";
   }
   cout << endl;
}

int main() {
   int arr[] = {12, 11, 13, 5, 6};
   int n = sizeof(arr) / sizeof(arr[0]);

   cout << "Original array: ";
   printArray(arr, n);
   insertionSort(arr, n);

   cout << "Sorted array: ";
   printArray(arr, n);

   return 0;}
```

Output:



Task No.: 04

Problem Statement: Write down a program that **multiplies two large non-negative integers** represented as strings. Because int and long int has memory limitations.

Source Code:

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
   string X, Y;
   while (cin >> X >> Y)
   {
      string product = "0";
      int lenX = X.length();
      int lenY = Y.length();
      string result(lenX + lenY, '0');

      for (int i = lenX - 1; i >= 0; --i)
      {
         for (int j = lenY - 1; j >= 0; --j)
         {
            int mul = (X[i] - '0') * (Y[j] - '0');
            int sum = mul + (result[i + j + 1] - '0');
            result[i + j + 1] = (sum % 10) + '0';
```

```
            result[i + j] += (sum / 10);
        }
    }
    int start = 0;
    while (start < result.length() && result[start] == '0')
    {
        start++;
    }
    if (start == result.length())
    {
        cout << "0";
    }
    else
    {
        for (int i = start; i < result.length(); i++)
        {
            cout << result[i];
        }
    }
    cout << endl;
  }

  return 0;
}
```

Output: