Muhammad Athar Abbas
SP25 - BSE - 082
Section A
Lab Assignment 3

Task 1:

TASKS:

1. Define a function **hypotenuse** that calculates the hypotenuse of a right triangle when the other two sides are given. The function should take two double arguments and return the hypotenuse as a double. Use this function in a program to determine the hypotenuse for each of the triangles shown below.

Triangle	Side 1	Side 2
1	3.0	4.0
2	5.0	12.0
3	8.0	15.0

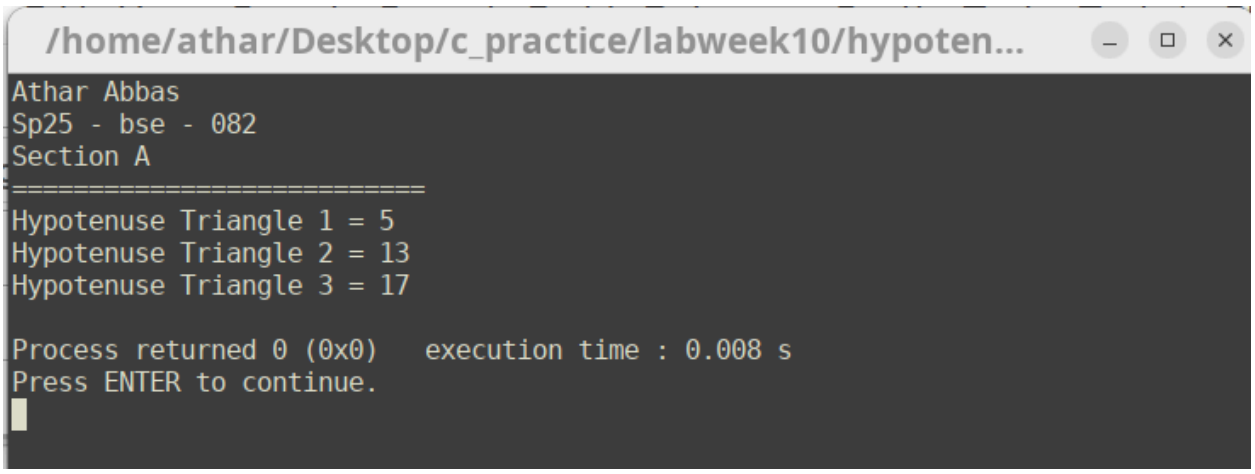
```
Let a = side of right triangle
    b = side of right triangle
    c = hypotenuse
Formula:  c^2 = a^2 + b^2
```

Solution:

```
#include <iostream>
#include <iomanip>
#include <cmath>
using namespace std;
double hypotenuse (double side1 , double side2);
int main() {
cout << "Athar Abbas" << endl;
cout << "Sp25 - bse - 082" << endl;
cout << "Section A" << endl;
cout << setw(27) << setfill('=') << '=' << endl;
cout << "Hypotenuse Triangle 1 = " << hypotenuse(3.0 , 4.0) <<
endl;
```

```
cout << "Hypotenuse Triangle 2 = " << hypotenuse(5.0 , 12.0) <<
endl;
cout << "Hypotenuse Triangle 3 = " << hypotenuse(8.0 , 15.0) <<
endl;
return 0;
}
double hypotenuse(double side1 , double side2){
return sqrt(side1 * side1 + side2 * side2);
}
```

Output:



Task 2:

2. Write a function to accept marks of a student (between 0-100) and return the grade according to the following criteria.

0 - 49	F
50 – 60	E
61 – 70	D
71 – 80	C
81 – 90	B
91 – 100	A

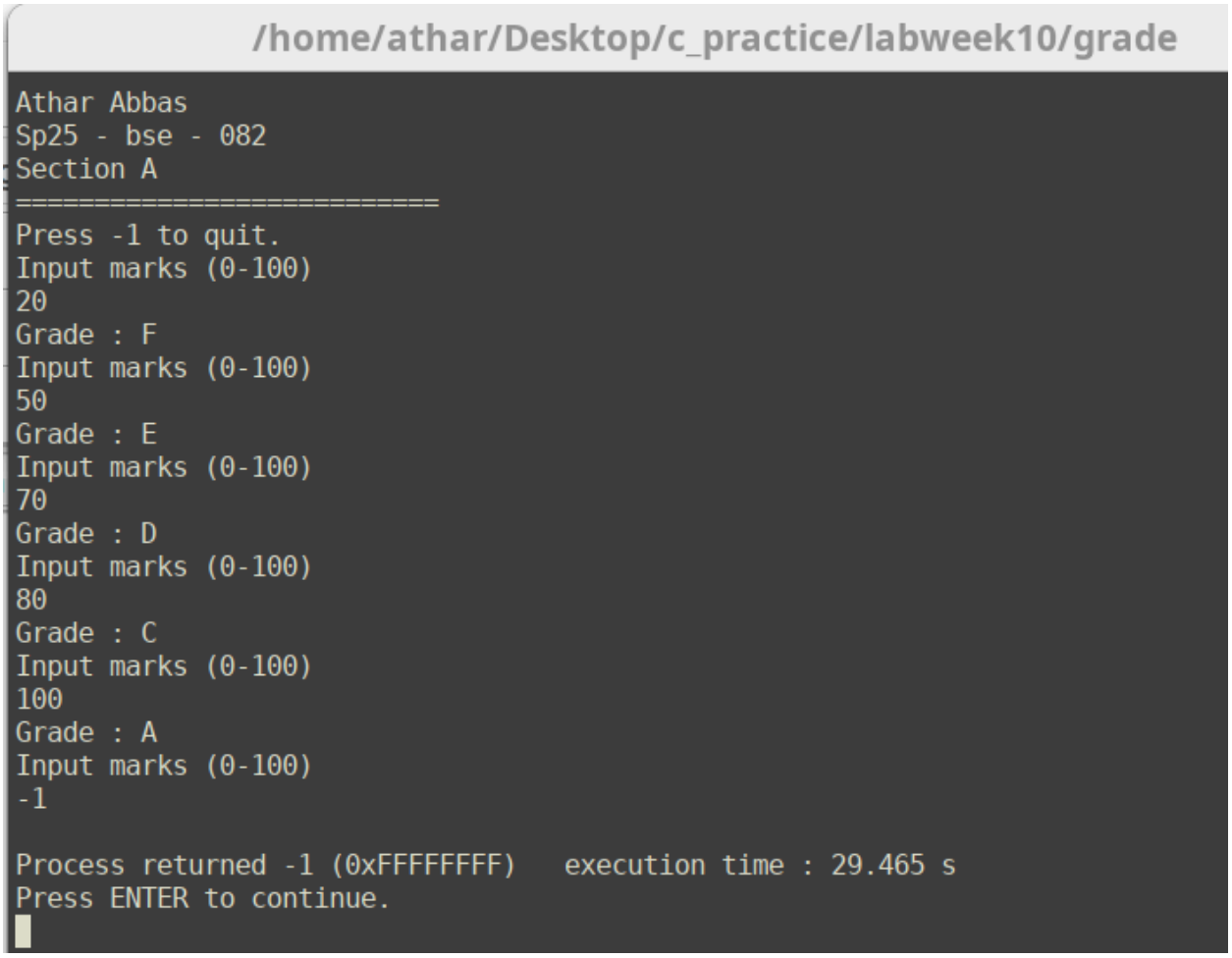
Solution:

```
#include <iostream>
#include <iomanip>
using namespace std;
char grade(int marks);
int main() {
cout << "Athar Abbas" << endl;
cout << "Sp25 - bse - 082" << endl;
cout << "Section A" << endl;
cout << setw(27) << setfill('=') << '=' << endl;
cout << "Press -1 to quit." << endl;
int num;
while (num != -1) {
cout << "Input marks (0-100)" << endl;
cin >> num;
    while (num > 100 || num < -1) {
        cin.clear();
```

```
        cin.ignore();
        cout << "Input marks (0-100)" << endl;
        cin >> num;
    }
cout << grade(num) << endl;
} ;
return 0;
}
char grade(int marks) {

    if (marks >= 0 && marks <= 49 ) {
        return 'F';
    }
    else if (marks >= 50 && marks <= 60){
        return 'E';
    }
    else if (marks >= 61 && marks <= 70) {
        return 'D';
    }
    else if (marks >= 71 && marks <= 80) {
        return 'C';
    }
    else if (marks >= 81 && marks <= 90) {
        return 'B';
    }
    else if (marks >= 91 && marks <= 100){
        return 'A';
    };
}
```

Output:



Task 3:

3. *(Square of Asterisks)* Write a function that displays at the left margin of the screen a solid square of asterisks whose side is specified in integer parameter side. For example, if side is 4, the function displays the following:

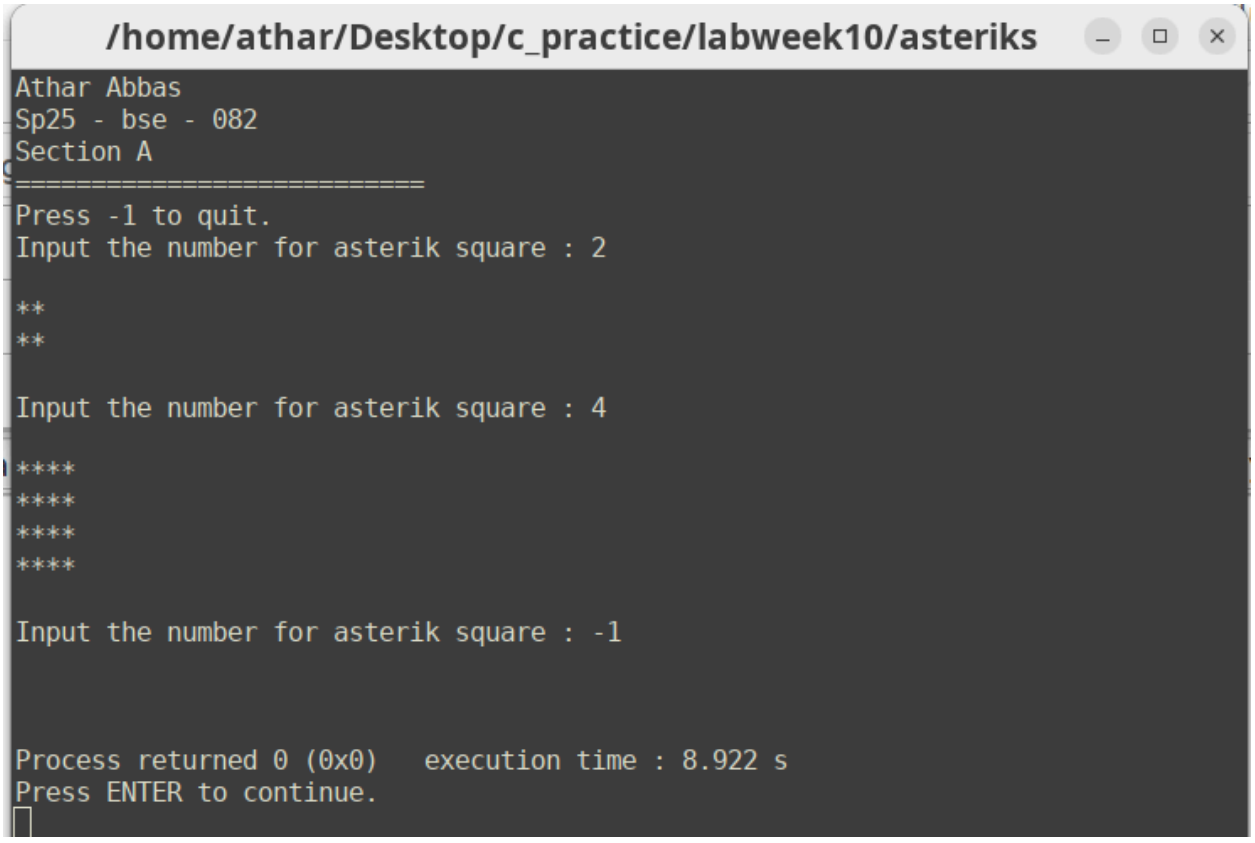
```
****
****
****
****
```

Solution:

```
#include <iostream>
#include <iomanip>
using namespace std;
void asteriksquare(int num);
int main() {
cout << "Athar Abbas" << endl;
cout << "Sp25 - bse - 082" << endl;
cout << "Section A" << endl;
cout << setw(27) << setfill('=') << '=' << endl;
cout << "Press -1 to quit." << endl;
int num;
do{
cout << "Input the number for asterik square : ";
cin >> num;
cout << endl;
asteriksquare(num);
cout << endl;
} while (num != -1);
return 0;
}
void asteriksquare(int num) {

    int i = 0;
    int j = 0;
    for (i ; i < num ; i++) {
        for(j = 0; j < num ; j++) {
                cout << '*';
        }
        cout << endl;
    }
}
```

Output:



Task 4:

4. Write a function, which inputs two characters from the user and compares them using a function named 'compareChars'. The function works as follows.
The 'compareChars' function receives 2 Input characters c1 and c2 as formal parameters, and returns following values:

i. Returns 1, if c1 is larger than c2 character in alphabetical order.
ii. Returns -1, if c1 is smaller than c2 character in alphabetical order.
iii. Returns 0, if c1 is equal to c2.
iv. Returns -99, if either c1 or c2 is not an alphabet.

(Hint: the characters are compared using their ASCII values).
Note: The function compareChar should ignore the alphabet's case, i.e., `'n'` and `'N'` must be treated as the same.
Call this function inside main using different characters as actual arguments and verify the function output by displaying its returned value on screen.

Solution:

```
#include <iostream>
#include <iomanip>
using namespace std;
int comparechars(char a , char b);
int main() {
cout << "Athar Abbas" << endl;
cout << "Sp25 - bse - 082" << endl;
cout << "Section A" << endl;
cout << setw(27) << setfill('=') << '=' << endl;
cout << endl;
cout << "Press '=' to quit. " << endl << endl;
char char1;
char char2;
do{
cout << endl;
cout << "Input first character ";
cin >> char1;
cout << "Input second character ";
cin >> char2;
cout << endl;
cout << comparechars(char1,char2) << endl;
} while (char1 != '=');
return 0;
}
int comparechars(char a , char b){
    if ((int(tolower(a)) < 97) || (int((tolower(a)) > 122 )) ||
((int(tolower(b))) < 97 || (int(tolower(a)) > 122))){
        return 99;
    }
    else if (tolower(a) > tolower(b)) {
        return 1;
    }
    else if (tolower(a) == tolower(b)) {
        return 0;
    }
```

```
    else if (tolower(b) > tolower(a)) {
        return -1;
}
    }
```

Output: