Task 1:

#include <iostream>

const int MAX\_ROWS = 3;

const int MAX\_COLUMNS = 2;

using namespace std;

int FindEven(int A[3][2], int length, int width);

int main()

{

int A[MAX\_ROWS][MAX\_COLUMNS] = { { 3, 2 },{ 4 , 5 },{ 2 , 2 } };

int length = 0;

int width = 0;

cout << FindEven(A, length, width);

return 0;

}

int FindEven(int A[3][2], int length, int width)

{

int counter = 0;

for (length = 0; length < 3; length++)

for (width = 0; width < 2; width++)

{

int even = A[length][width] % 2;

if (even == 0)

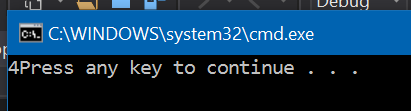
counter++;

}

return counter;

}

Output:



Task 2:

#include <iostream>

const int NUM\_DEPTS = 2;

const int NUM\_STORES = 2;

const int NUM\_MONTHS = 12;

using namespace std;

float Sale[NUM\_STORES][NUM\_MONTHS][NUM\_DEPTS];

// 2 12 2

/\*

0 0 0 = 1.1

0 0 1 = 1.2

\*/

int main()

{

int InputMonth;

float Sale[NUM\_STORES][NUM\_MONTHS][NUM\_DEPTS] =

{ 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0, 2.1, 2.2,

2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0, 3.1, 3.2,

3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 4.0, 4.1, 4.2,

2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7, 2.8, 2.9, 3.0, 3.1, 3.2

};

cout << "Enter the month. " << endl;

cin >> InputMonth;

for (int j = 0; j < NUM\_STORES; j++)

for (int i = 0; i < NUM\_DEPTS; i++)

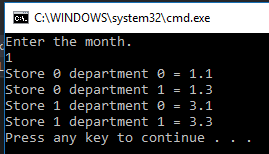
cout << "Store " << j << " department " << i << " = " <<

Sale[j][i][InputMonth - 1] << endl;

return 0;

}

Output:



Task 3:

#include<iostream>

using namespace std;

//Example 11.5.2: Recursive function: Writing the recursive case.

int Combinations(int Y, int X) {

cout << " Y: " << Y << " X: " << X << endl;

if (X == 1)

return Y;

if (X == Y)

return 1;

else if ((Y > X) && X>1)

return Combinations(Y - 1, X - 1) + Combinations(Y - 1, X);

else return 0;

}

int main() {

int YY = 0;

int XX = 0;

cout << "Enter Y & X" << endl;

cin >> YY;

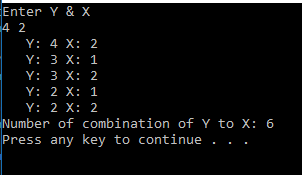
cin >> XX;

cout << "Number of combination of Y to X: " << Combinations(YY, XX) << endl; // Input X & Y

return 0;

}

Output:



Task 4:

#include<iostream>

using namespace std;

//Example 11.5.2: Recursive function: Writing the recursive case.

int Combinations(int Y, int X) {

cout << " Y: " << Y << " X: " << X << endl;

if (X == 1)

return Y;

if (X == Y)

return 1;

else if ((Y > X) && X>1)

return Combinations(Y - 1, X - 1) + Combinations(Y - 1, X);

else return 0;

}

int main() {

cout << "The number of combinations of 4 items made out of a total of 8 items is: " << Combinations(8, 4) << endl; // Input X & Y

return 0;

}

Output:



Task 5:

#include<iostream>

using namespace std;

//Example 11.5.2: Recursive function: Writing the recursive case.

int Combinations(int Y, int X, int level) {

cout << "Recursive level: " << level << endl;

cout << " Y: " << Y << " X: " << X << endl;

if (X == 1)

return Y;

if (X == Y)

return 1;

else if ((Y > X) && X>1)

return Combinations(Y - 1, X - 1, level + 1) + Combinations(Y - 1, X, level + 1);

else return 0;

}

int main() {

cout << "Combination(8, 4, 1) = " << Combinations(8, 4, 1) << endl; // Input X & Y

return 0;

}

Output:

