Instructions: Please read carefully

- Please rename this file as only your ID number **(e.g. 18-*****-1.doc or 18-*****-1.pdf).**
- Submit the file before **deadline** in the Portal Lab Performance section labeled **Lab task 3.** **If you cannot complete the full task, do not worry. Just upload what you have completed.**

1. Find the summation of the boundary elements for the given array. Take input from user keyboard.

1	2	3	4	5
14	15	16	17	6
13	20	19	18	7
12	11	10	9	8

For example,

Matrix_1:

1 2 3 4 5

14 15 16 17 6

13 1 9 18 7

12 11 10 9 8

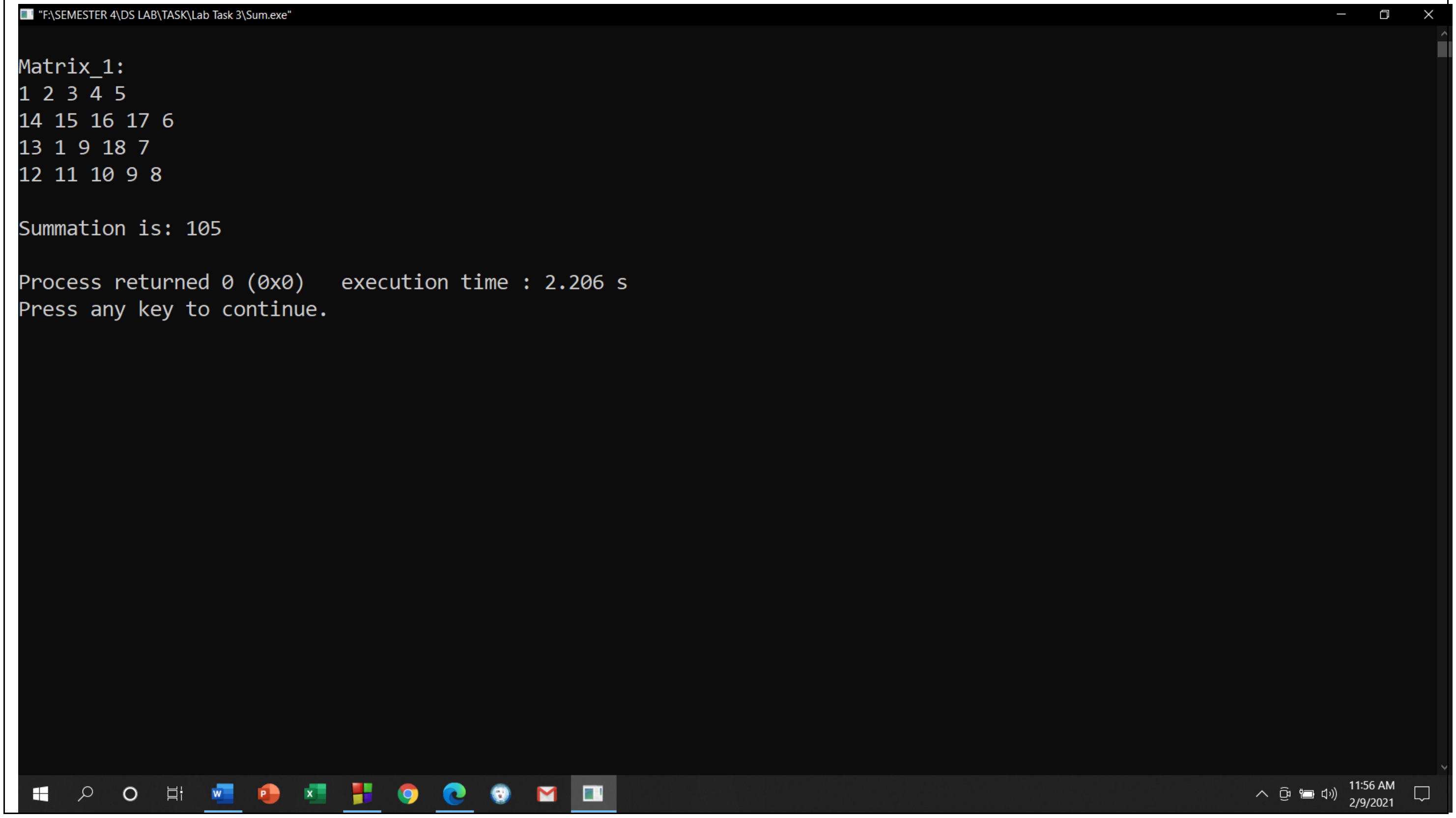
Output:

Summation is: 105

Your code here:

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

int main()
{
   int r=4, d=5, a[r][d], sum=0, i, j;
 cout << endl << "Matrix_1: " << endl;
   for(i = 0; i < r; ++i)
   {
       for(j = 0; j < d; ++j)
      {
         cin >> a[i][j];

      }
   }

   for (int i = 0; i < r; i++) {
      for (int j = 0; j < d; j++)
     {
        if (i == 0 || j == 0 || i == r - 1 || j == d - 1)
        {
            sum+=a[i][j];
        }
     }
   }
cout<<"\nSummation is: "<<sum;
cout<<endl;

   return 0;
}
```

Your whole Screenshot here: (Console Output):



```
Matrix_1:
1 2 3 4 5
14 15 16 17 6
13 1 9 18 7
12 11 10 9 8

Summation is: 105

Process returned 0 (0x0)   execution time : 2.206 s
Press any key to continue.
```

2. Find the summation of the diagonal and anti-diagonal elements for the given array. Take input from user keyboard.

1	2	3	4	5
14	15	16	17	6
13	20	19	18	7
12	11	10	9	8
21	22	23	24	25

For example,
Matrix_1:

1 2 3 4 5
14 15 16 17 6
13 1 9 18 7
12 11 10 9 8
21 22 23 24 25

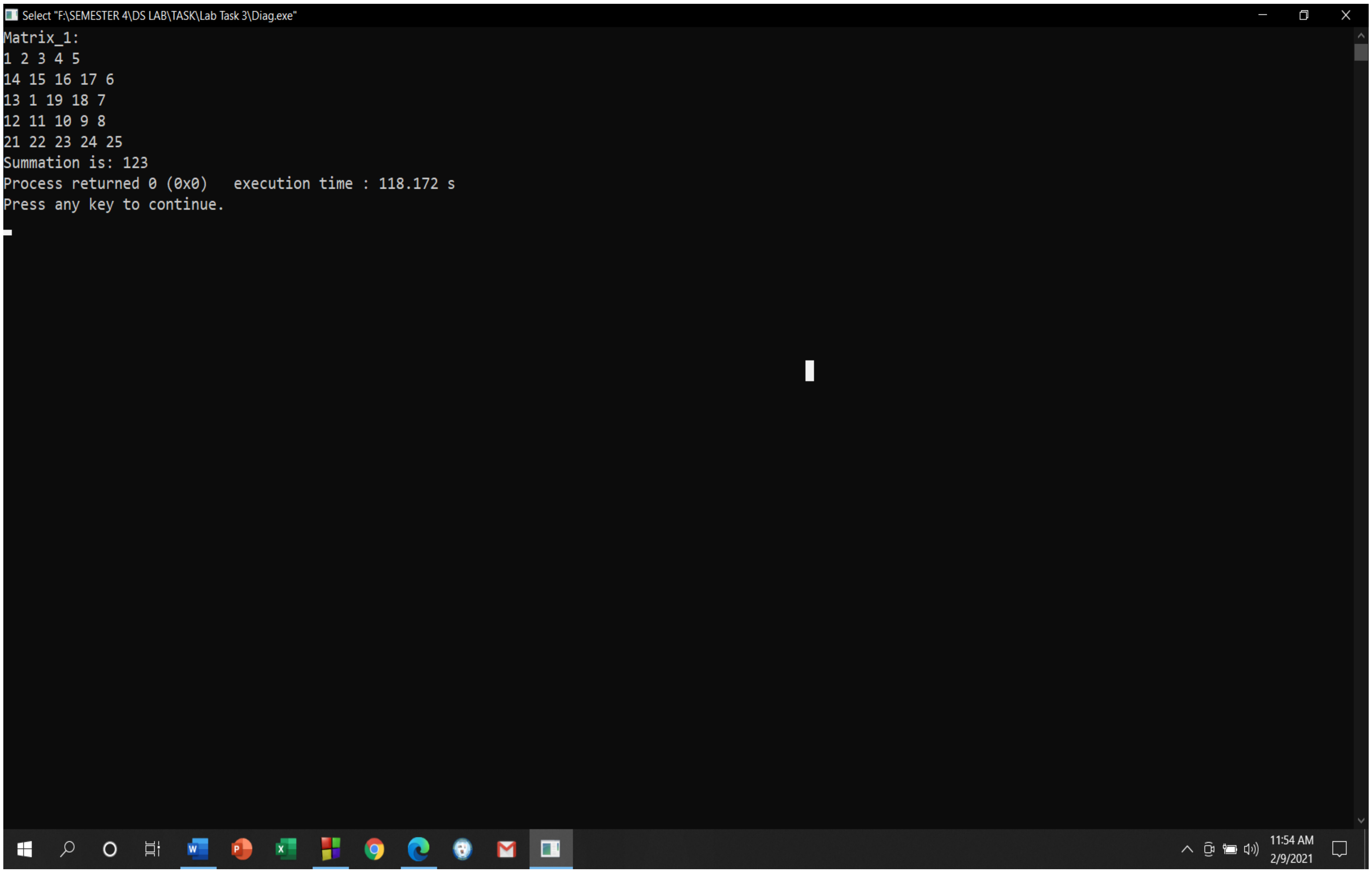
Output:

Summation is: 123

Your code here:

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

int main()
{
   int r=5, d=5, a[r][d], sum=0, i, j;


 cout << "Matrix_1: " << endl;
   for(i = 0; i < r; ++i)
   {
        for(j = 0; j < d; ++j)
      {
          cin >> a[i][j];


      }
   }

   for (int i = 0; i < r; i++) {
       for (int j = 0; j < d; j++)
     {
         if (i == j || (i + j) == (d - 1))
         {
             sum+=a[i][j];
         }

     }
   }

cout<<"Summation is: "<<sum;

   return 0;
}
```

Your whole Screenshot here: (Console Output):



```
Matrix_1:
1 2 3 4 5
14 15 16 17 6
13 1 19 18 7
12 11 10 9 8
21 22 23 24 25
Summation is: 123
Process returned 0 (0x0)   execution time : 118.172 s
Press any key to continue.
```

3. Write a code that will create custom ciphers (encoded words) on strings. Follow this procedure:
   1. Write a function named ***encode*** that takes TWO parameters, a string **s** and an integer **j**.
   2. Skip **j** number of characters in the string and increase the ASCII value of the next character by 2.
   3. Perform step **(ii)** throughout the string.
   4. Return the converted string from **encode** function.

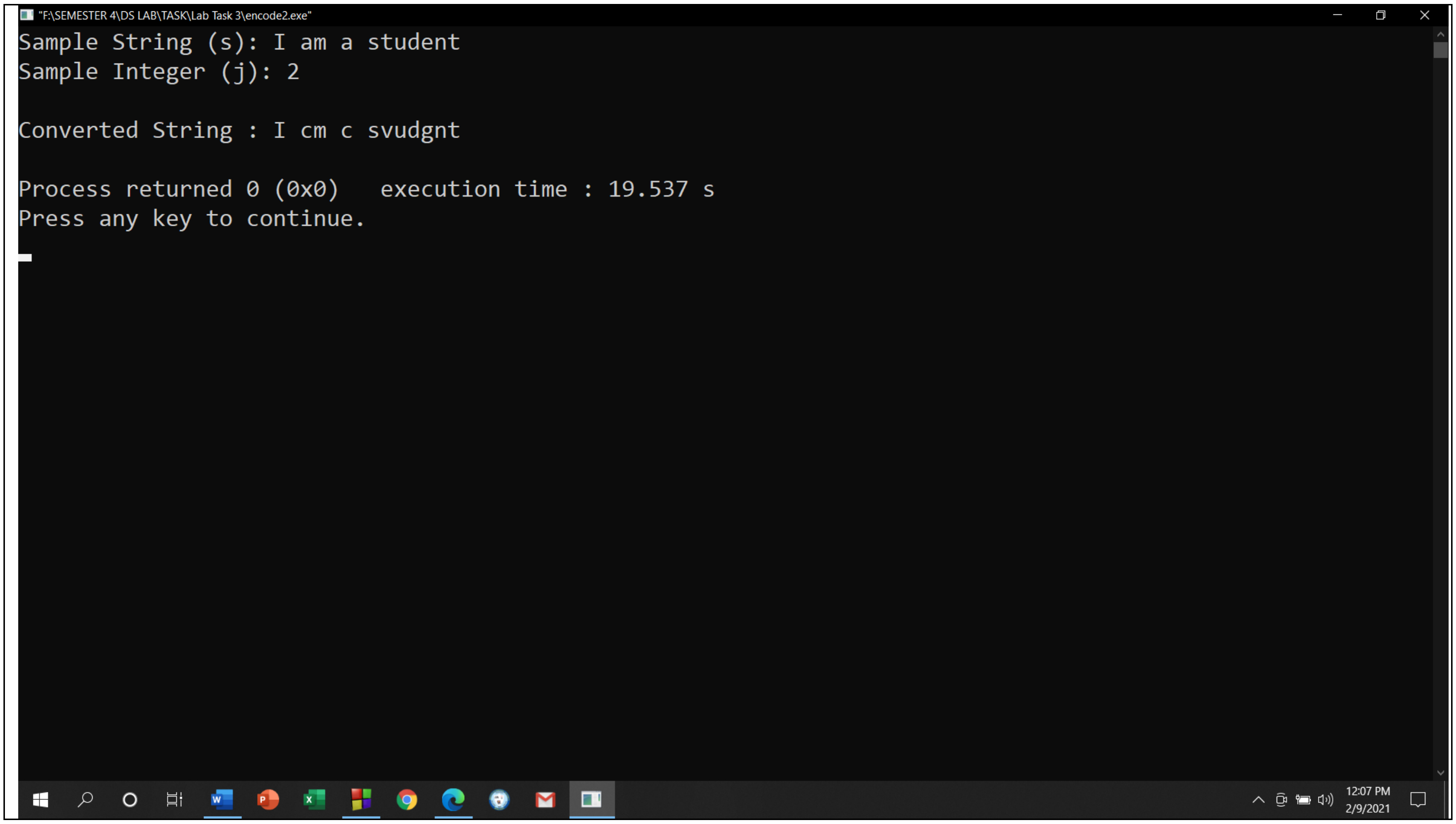
For example,

Sample String (s): I am a student
Sample Integer (j): 2
Converted String: I cm c svudgnt

Your code here:

```
#include <iostream>
using namespace std;
void encode(string s, int j)
{
int i;
for(i=j;i<s.length();i++)
{
s[i]=s[i]+2;
i+=j;
}
cout<<"\nConverted String : ";
for(i=0;i<s.length();i++)
{
cout<<s[i];
}
}
int main()
{
int j;
string s;
  cout<<"Sample String (s): ";
  getline(cin,s);
  cout<<"Sample Integer (j): ";
  cin>>j;
 encode(s,j);
 cout<<endl;
return 0;
}
```

Your whole Screenshot here: (Console Output):



```
Sample String (s): I am a student
Sample Integer (j): 2

Converted String : I cm c svudgnt

Process returned 0 (0x0)   execution time : 19.537 s
Press any key to continue.
```

4. Write a program with appropriate data structure to keep records of 10 students. Each student will have the following information:
    1. Unique ID (you can use *integer* for this)
    2. Number of Credits Completed
    3. CGPA

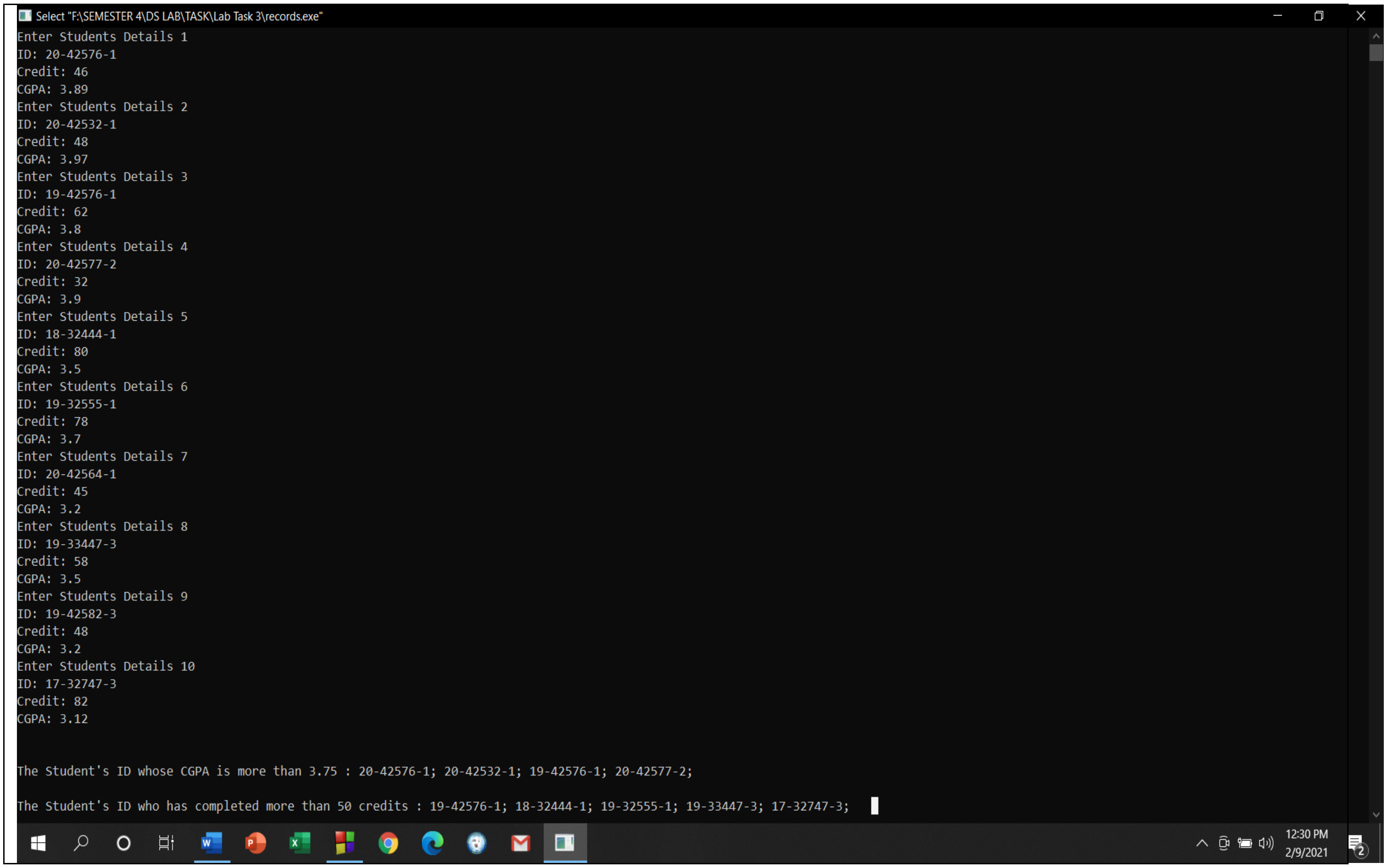
Print all the student's ID whose CGPA is more than **3.75**.

Print all the student's ID who has completed more than **50** credits.

Your code here:

```
#include <iostream>
using namespace std;
struct student
{
   string id;
   int cr;
   float cg;
};
int main()
{
  student arr[10];
  for(int i=0;i<10;i++)
  {
     cout<<"Enter Students Details "<<(i+1)<<endl;
     cout<<"ID: ";
     cin>>arr[i].id;
     cout<<"Credit: ";
     cin>>arr[i].cr;
     cout<<"CGPA: ";
     cin>>arr[i].cg;

  }
  cout<<"\n\nThe Student's ID whose CGPA is more than 3.75 : ";
  for(int i=0;i<10;i++)
  {
     if(arr[i].cg>3.75)
      cout<<arr[i].id<<"; ";

  }
  cout<<"\n\nThe Student's ID who has completed more than 50 credits : ";
  for(int i=0;i<10;i++)
  {
     if(arr[i].cr>50)
      cout<<arr[i].id<<"; ";

  }
cout<<endl;
}
```

Your whole Screenshot here: (Console Output):



```
Enter Students Details 1
ID: 20-42576-1
Credit: 46
CGPA: 3.89
Enter Students Details 2
ID: 20-42532-1
Credit: 48
CGPA: 3.97
Enter Students Details 3
ID: 19-42576-1
Credit: 62
CGPA: 3.8
Enter Students Details 4
ID: 20-42577-2
Credit: 32
CGPA: 3.9
Enter Students Details 5
ID: 18-32444-1
Credit: 80
CGPA: 3.5
Enter Students Details 6
ID: 19-32555-1
Credit: 78
CGPA: 3.7
Enter Students Details 7
ID: 20-42564-1
Credit: 45
CGPA: 3.2
Enter Students Details 8
ID: 19-33447-3
Credit: 58
CGPA: 3.5
Enter Students Details 9
ID: 19-42582-3
Credit: 48
CGPA: 3.2
Enter Students Details 10
ID: 17-32747-3
Credit: 82
CGPA: 3.12


The Student's ID whose CGPA is more than 3.75 : 20-42576-1; 20-42532-1; 19-42576-1; 20-42577-2;

The Student's ID who has completed more than 50 credits : 19-42576-1; 18-32444-1; 19-32555-1; 19-33447-3; 17-32747-3;
```