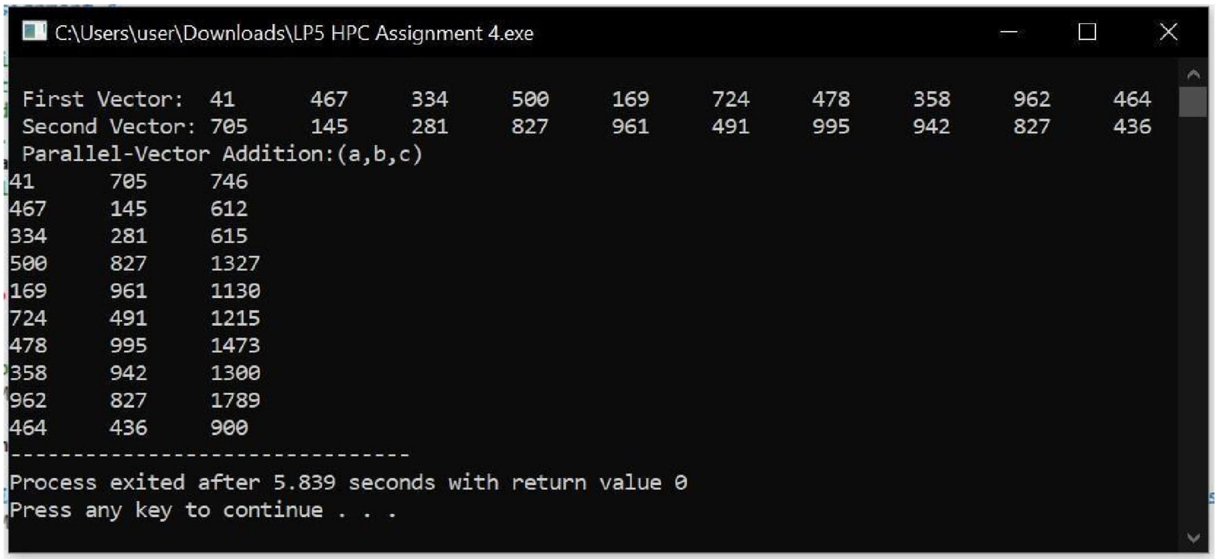
Experiement 4

A. VECTOR ADDITION USING CUDA C

```
#include<stdio.h>
#include<iostream>
#include<cstdlib>
#include<omp.h>         using
namespace std;  #define MAX
10

int main()
{
  int a[MAX],b[MAX],c[MAX],i;    printf("\n
First Vector:\t");
  //Instruct a master thread to fork and generate more threads to process following loop
structure      #pragma omp parallel for    for(i=0;i<MAX;i++)
  {
    a[i]=rand()%1000;
  }
  //Discuss issue of this for loop below-if we make it parallel, possibly values that get printed
will not be in sequence as we dont have any control on order of threads execution
for(i=0;i<MAX;i++)
  {
printf("%d\t",a[i]);
  }
  printf("\n Second Vector:\t");   #pragma
omp parallel for    for(i=0;i<MAX;i++)
  {
    b[i]=rand()%1000;
  }
  for(i=0;i<MAX;i++)
  {
    printf("%d\t",b[i]);
  }
  printf("\n Parallel-Vector Addition:(a,b,c)\t");
  #pragma omp parallel for    for(i=0;i<MAX;i++)
  {      c[i]=a[i]+b[i];
  }
  for(i=0;i<MAX;i++)
  {
    printf("\n%d\t%d\t%d",a[i],b[i],c[i]);
  }
}
```



```
 First Vector:  41      467     334     500     169     724     478     358     962     464
 Second Vector: 705     145     281     827     961     491     995     942     827     436
 Parallel-Vector Addition:(a,b,c)
41      705     746
467     145     612
334     281     615
500     827     1327
169     961     1130
724     491     1215
478     995     1473
358     942     1300
962     827     1789
464     436     900
--------------------------------
Process exited after 5.839 seconds with return value 0
Press any key to continue . . .
```

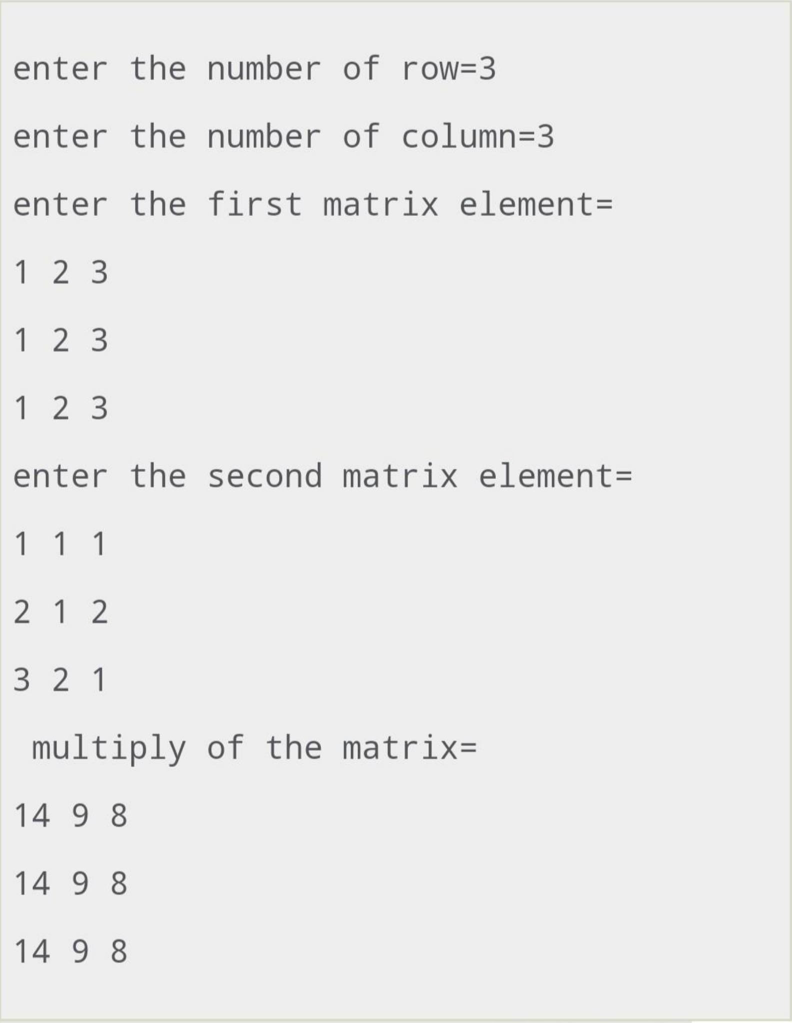
B. Matrix Multiplication

```
#include <iostíeam>
#include<cstdlib>
#include<omp.h>
using namespace std;
int main()
{
int a[10][10],b[10][10],mul[10][10],í,c,i,j,k;
cout<<"enteí the numbeí of íow=";
cin>>í;
cout<<"enteí the numbeí of column=";
cin>>c;
cout<<"enteí the fiíst matíix element=\n";foí(i=0;i<í;i++)
{
foí(j=0;j<c;j++)
{
cin>>a[i][j];
}
}
cout<<"enteí the second matíix element=\n";
foí(i=0;i<í;i++)
{
foí(j=0;j<c;j++)
```

```
{
cin>>b[i][j];
}
}
cout<<"multiply of the matíix=\n";
foí(i=0;i<í;i++)
{
foí(j=0;j<c;j++)
{
mul[i][j]=0;
foí(k=0;k<c;k++)
{
mul[i][j]+=a[i][k]*b[k][j];
}
}
}
//foí píinting íesult
foí(i=0;i<í;i++)
{
foí(j=0;j<c;j++)
{
cout<<mul[i][j]<<"  ";
}
cout<<"\n";
```

```
}
íetuín 0;
}
```

Output:-



```
enter the number of row=3
enter the number of column=3
enter the first matrix element=
1 2 3
1 2 3
1 2 3
enter the second matrix element=
1 1 1
2 1 2
3 2 1
 multiply of the matrix=
14 9 8
14 9 8
14 9 8
```