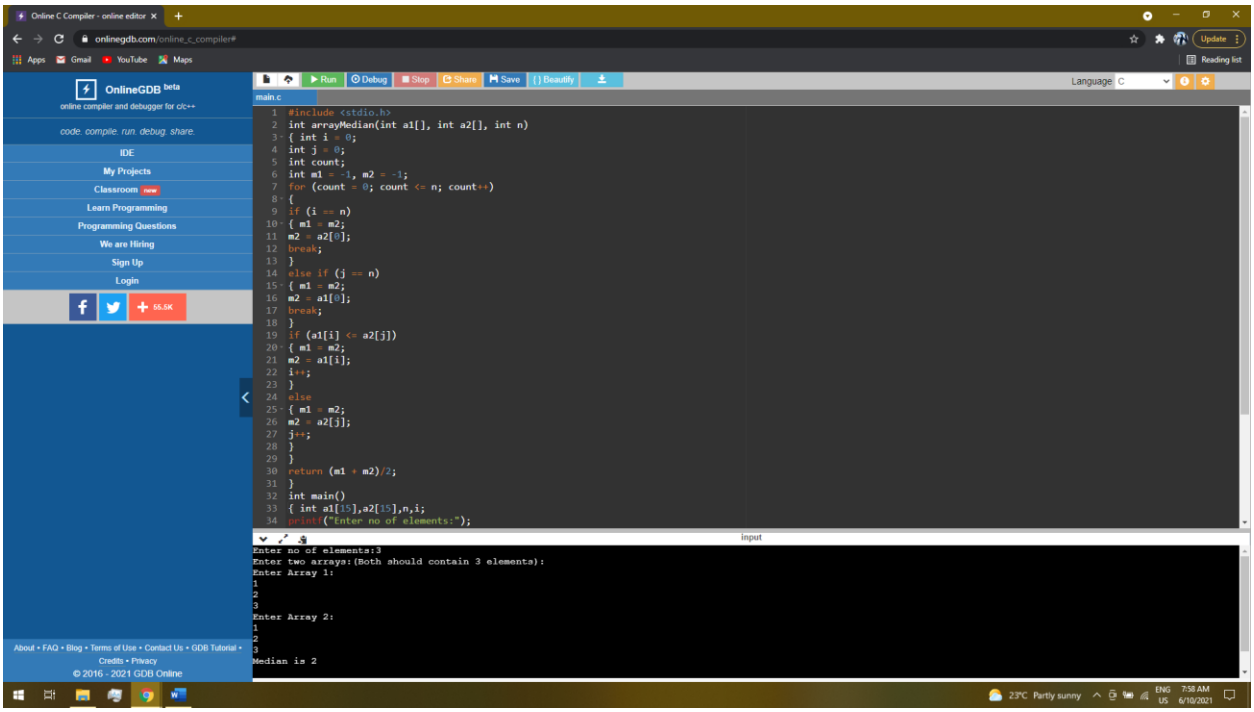
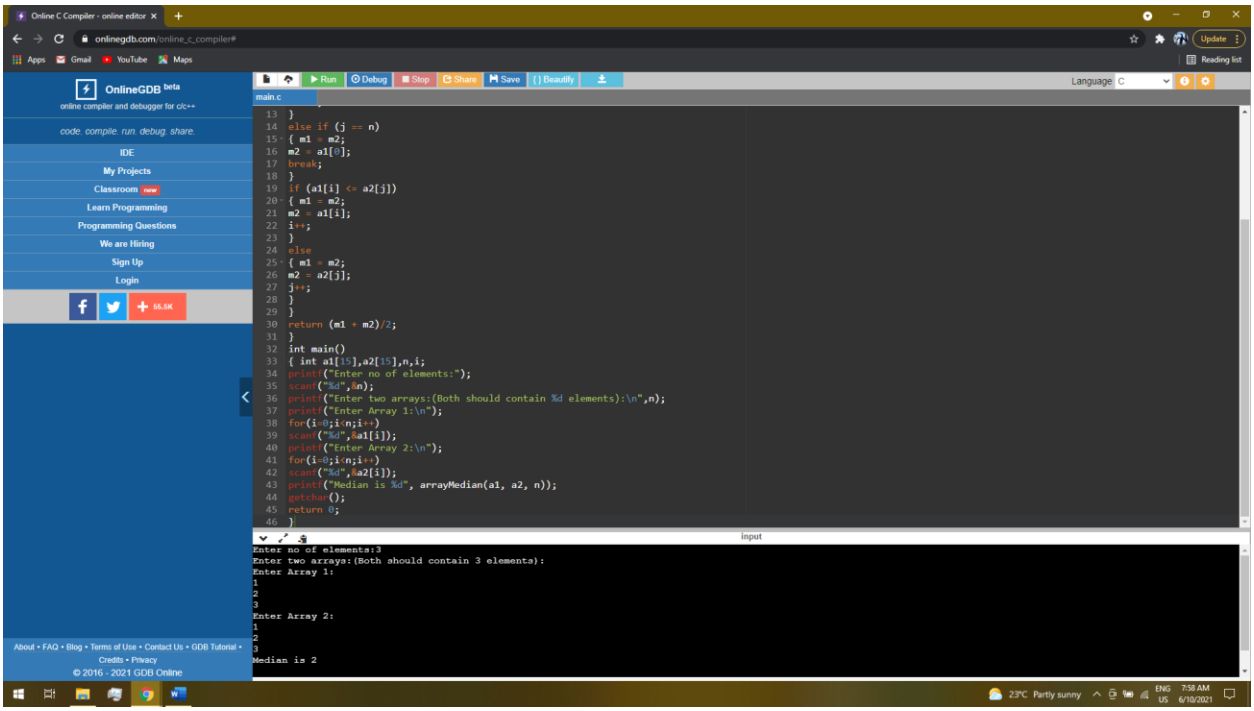
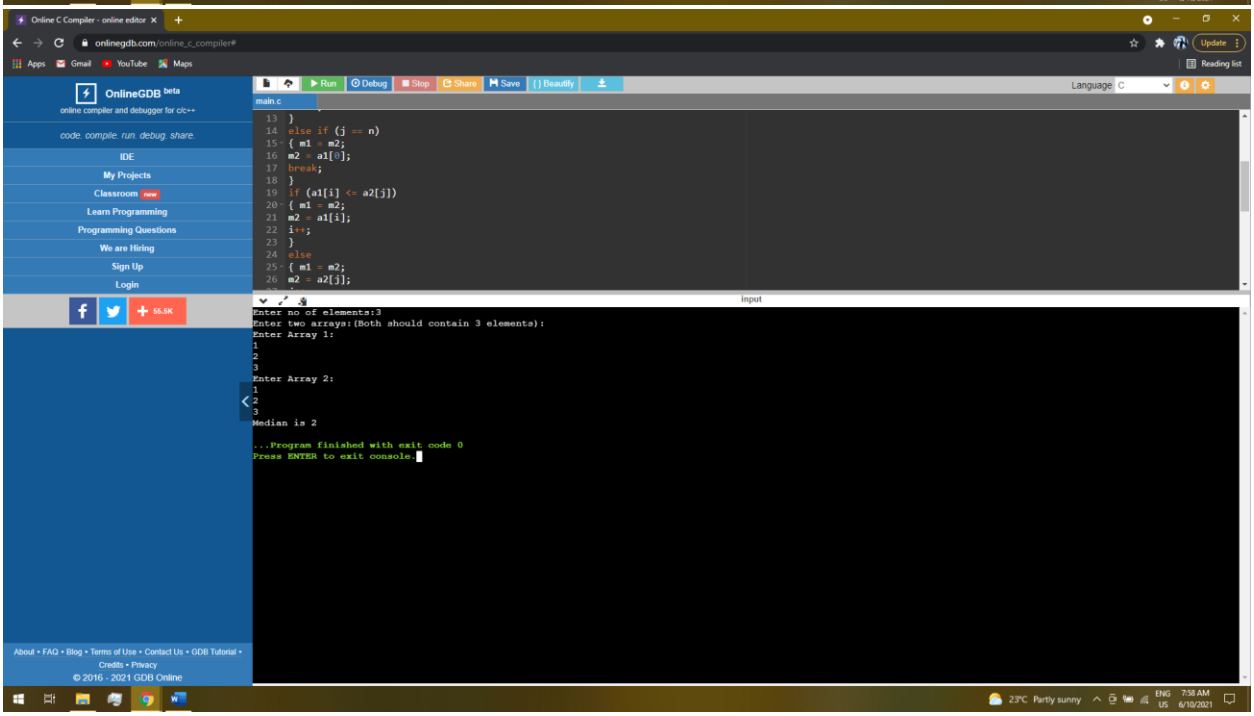
OUTPUT FOR MEDIAN







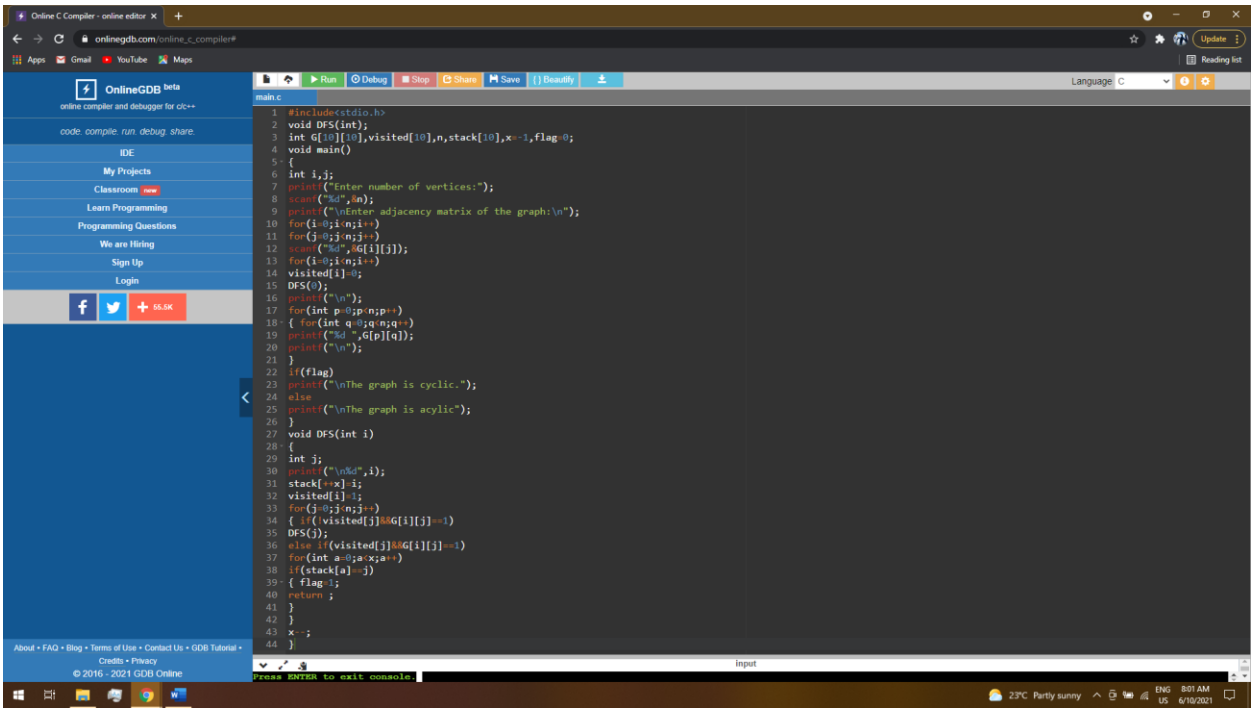
CODE FOR MEDIAN

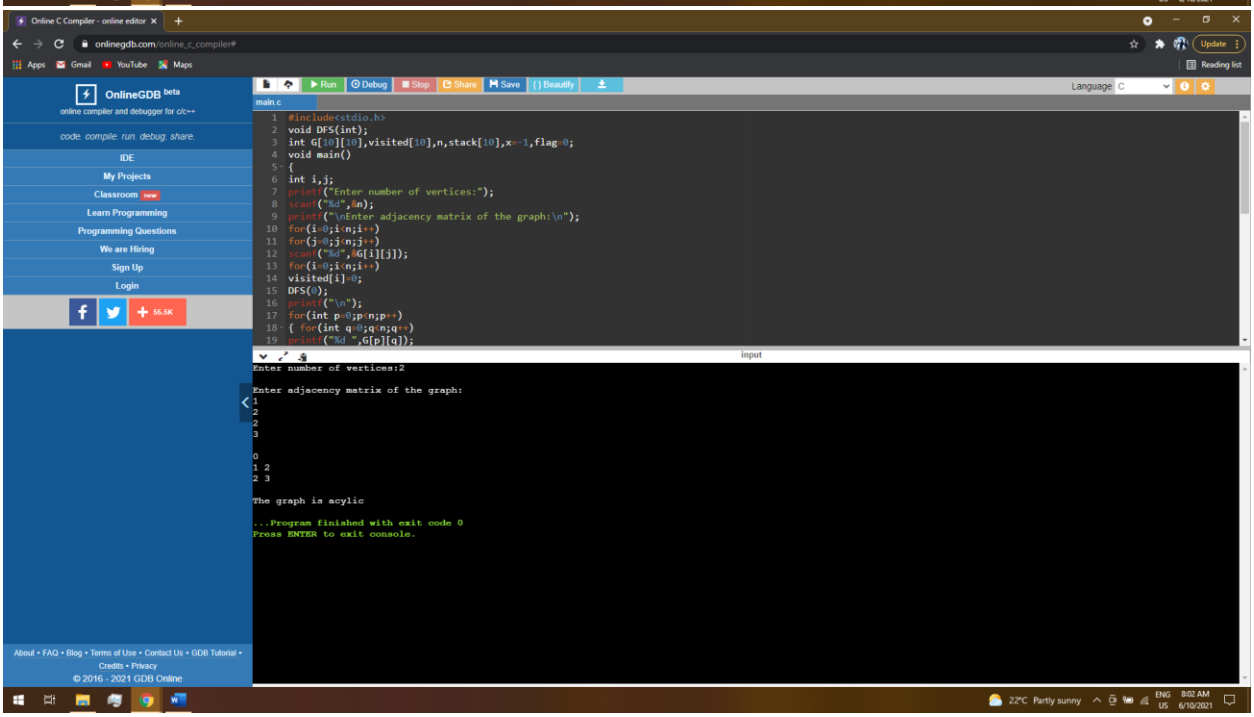
```
#include <stdio.h>

int arrayMedian(int a1[], int a2[], int n)

{ int i = 0;

int j = 0;
```

```
int count;
int m1 = -1, m2 = -1;
for (count = 0; count <= n; count++)
{
if (i == n)
{ m1 = m2;
m2 = a2[0];
break;
}
else if (j == n)
{ m1 = m2;
m2 = a1[0];
break;
}
if (a1[i] <= a2[j])
{ m1 = m2;
m2 = a1[i];
i++;
}
else
{ m1 = m2;
m2 = a2[j];
j++;
}
}
return (m1 + m2)/2;
}
int main()
{ int a1[15],a2[15],n,i;
```

```
printf("Enter no of elements:");
scanf("%d",&n);
printf("Enter two arrays:(Both should contain %d elements):\n",n);
printf("Enter Array 1:\n");
for(i=0;i<n;i++)
scanf("%d",&a1[i]);
printf("Enter Array 2:\n");
for(i=0;i<n;i++)
scanf("%d",&a2[i]);
printf("Median is %d", arrayMedian(a1, a2, n));
getchar();
return 0;
}
```

OUTPUT FOR DFS CYCLIC





CODE FOR DFS CYCLIC

```
#include<stdio.h>

void DFS(int);

int G[10][10],visited[10],n,stack[10],x=-1,flag=0;
```

```
void main()
{
int i,j;
printf("Enter number of vertices:");
scanf("%d",&n);
printf("\nEnter adjacency matrix of the graph:\n");
for(i=0;i<n;i++)
for(j=0;j<n;j++)
scanf("%d",&G[i][j]);
for(i=0;i<n;i++)
visited[i]=0;
DFS(0);
printf("\n");
for(int p=0;p<n;p++)
{ for(int q=0;q<n;q++)
printf("%d ",G[p][q]);
printf("\n");
}
if(flag)
printf("\nThe graph is cyclic.");
else
printf("\nThe graph is acylic");
}
void DFS(int i)
{
int j;
printf("\n%d",i);
stack[++x]=i;
visited[i]=1;
```

```
for(j=0;j<n;j++)
{ if(!visited[j]&&G[i][j]==1)
DFS(j);
else if(visited[j]&&G[i][j]==1)
for(int a=0;a<x;a++)
if(stack[a]==j)
{ flag=1;
return ;
}
}
x--;
}
```