```
#include <iostream>
#include <sys/shm.h>
#include <string>
using namespace std;


struct Student
{

  int rollNumber;
  char name[20];



};

int main()
{

 int key=shmget(12320, 1024, IPC_CREAT | IPC_EXCL | 0666); //12321 is the user key// 1024
is the bytes we want to use as shared memory

 Student* ptr= (Student*) shmat(key, NULL, 0); //this call will attach the shared region to this
process's address space, and return the  pointer to it. the original pointer returned is void* we
can typecast it to any kind of pointer.

 cout<<"shm created with id= "<<key<<endl;
 ptr[0].rollNumber=-1;

 cout<<"Waiting for the client process to send data.!"<<endl;
 while (ptr[0].rollNumber==-1); //busy wait

 cout<<endl;
 for (int i=1; i<= ptr[0].rollNumber; i++)
 {

   cout<<"Roll Number: "<<ptr[i].rollNumber<<endl;
   cout<<"Name: "<<ptr[i].name<<endl;
   cout<<endl<<endl;
 }


 //detach
 shmdt(ptr);
shmctl(key, IPC_RMID, NULL); //mark the shared region for deletion

}
```