

## Task 1

```
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <wait.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{
    int fd[2];
    // fd[0] - read
    // fd[1] - write
    if (pipe(fd) == -1)
    {
        printf("An error occurred with opening the pipe\n");
        return 1;
    }
    int id = fork();
    if (id == 0)
    {
        close(fd[0]);
        int x;
        printf("Input a number: ");
        scanf("%d", &x);
        if (write(fd[1], &x, sizeof(int)) == -1)
        {
            printf("Error Occured While Writing");
            return 2;
        }
        sleep(1);
        printf("Input another number: ");
        scanf("%d", &x);
        if (write(fd[1], &x, sizeof(int)) == -1)
        {
            printf("Error Occured While Writing");
            return 2;
        }
        sleep(1);
        printf("Input another number: ");
        scanf("%d", &x);
        if (write(fd[1], &x, sizeof(int)) == -1)
        {
```

```
        printf("Error Occured While Writing");
        return 2;
    }
    close(fd[1]);
}
else
{

    close(fd[1]);
    int y;
    if (read(fd[0], &y, sizeof(int)) == -1)
    {

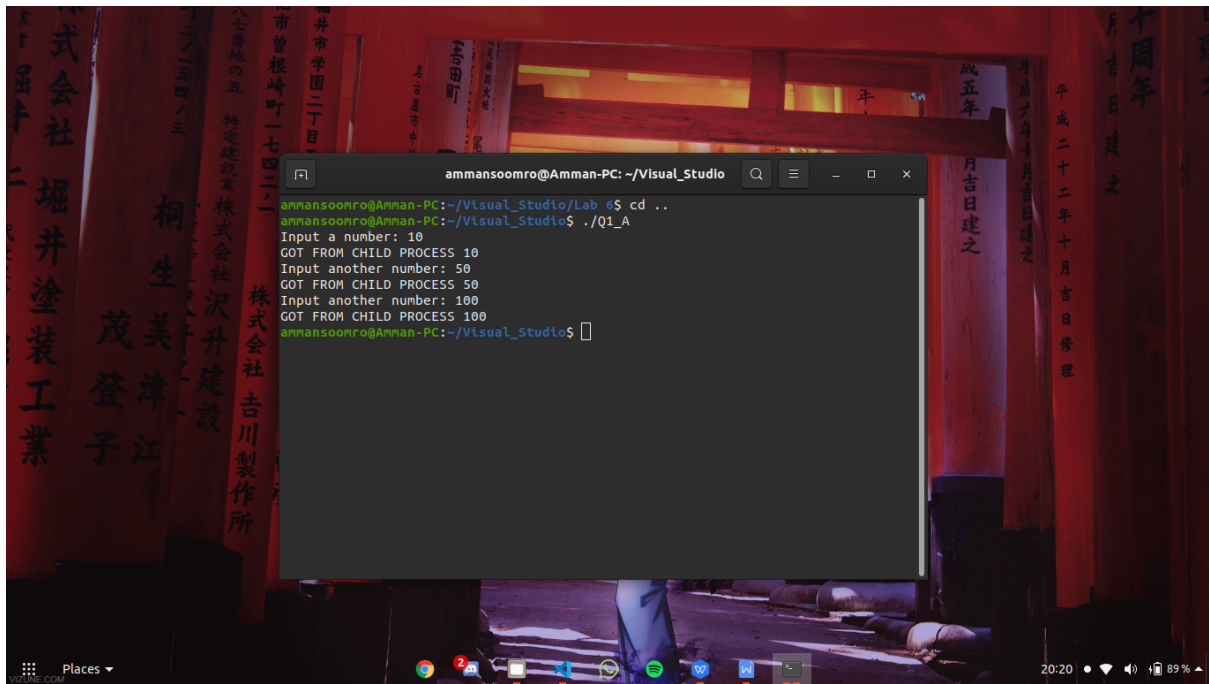
        printf("Error Occured While reading");
        return 2;
    }
    printf("GOT FROM CHILD PROCESS %d \n", y);
    if (read(fd[0], &y, sizeof(int)) == -1)
    {

        printf("Error Occured While reading");
        return 2;
    }
    printf("GOT FROM CHILD PROCESS %d \n", y);
    if (read(fd[0], &y, sizeof(int)) == -1)
    {

        printf("Error Occured While reading");
        return 2;
    }
    printf("GOT FROM CHILD PROCESS %d \n", y);
    close(fd[0]);

}
return 0;
}
```

K191048  
Amman Soomro



## Task 2

Read

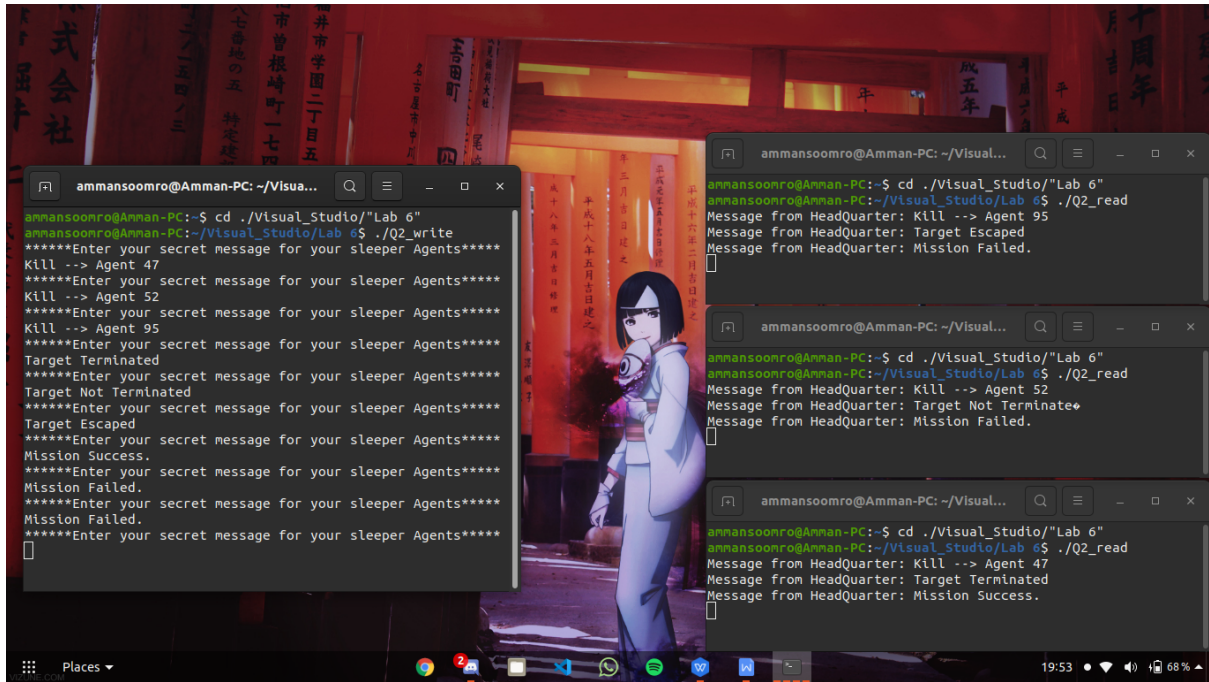
```
#include<stdio.h>
#include<sys/stat.h>
#include<sys/types.h>
#include<fcntl.h>
#include<unistd.h>
#include<string.h>
int main(void) {
    int fd, retval;
    char buffer[20];
    fd = open("/tmp/ammansoomro",O_RDONLY);
    while(1){
        fflush(stdin);
        memset(buffer, 0, sizeof(buffer));
        read(fd, buffer, sizeof(buffer));
        if(buffer[0]=='0')
        {
            break;
        }
        printf("Message from HeadQuarter: ");
        puts(buffer);
    }
    close(fd);
}
```

```
    return 0;  
}
```

## Write

```
#include<stdio.h>  
#include<sys/stat.h>  
#include<sys/types.h>  
#include<fcntl.h>  
#include<unistd.h>  
#include<string.h>  
int main(void) {  
    int fd, retval;  
    char buffer[20];  
    retval = mkfifo("/tmp/ammansoomro",0777);  
    fd = open("/tmp/ammansoomro",O_WRONLY);  
    while(1){  
        fflush(stdin);  
        memset(buffer, 0, sizeof(buffer));  
        printf("*****Enter your secret message for your sleeper  
Agents*****\n");  
        gets(buffer);  
        write(fd,buffer,sizeof(buffer));  
        if(buffer[0]=='0')  
        {  
            break;  
        }  
    }  
    close(fd);  
    return 0;  
}
```

K191048  
Amman Soomro



## Task 3

### File A

```
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
void main(){
    int num;
    key_t key = ftok("key_file",10);
    int shmid = shmget(key,1024,0666|IPC_CREAT);
    int *str = shmat(shmid,(void*)0,0);
    printf("Enter the number(1-10):");
    scanf("%d",&num);
    *str=num;
    printf("Data written in memory: %d\n",*str);
    while(*str==num);
    while(1){
        sleep(1);
        if(*str=='r'){
            printf("Data written in memory: %c\n", (char)*str);
            *str='*';}
    }
```

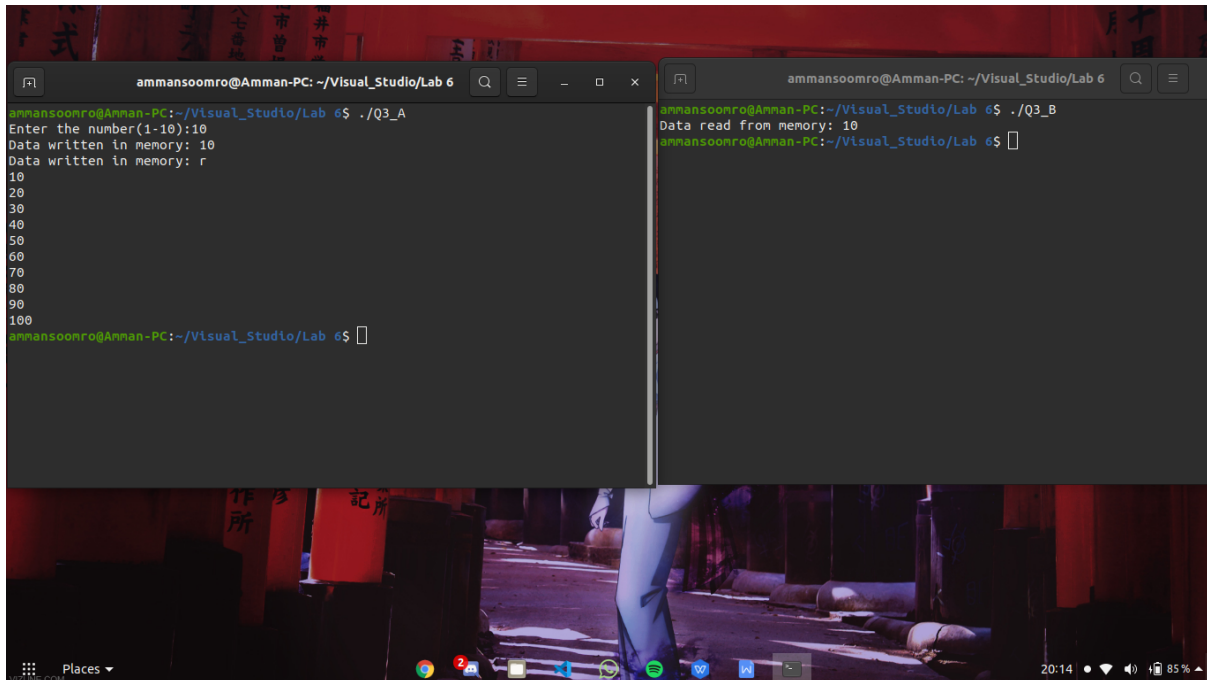
```
        else if(*str=='b')
            break;
        else if(*str!='*'){
            printf("%d",*str);
            printf("\n");
            *str='*';
        }
    }
    shmdt(str);
    shmctl(shmid,IPC_RMID,NULL);
    return;
}
```

## File B

```
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
void main()
{
    int num,i,res;
    key_t key = ftok("key_file",10);
    int shmid = shmget(key,1024,0666|IPC_CREAT);
    int *str = shmat(shmid,(void*)0,0);
    printf("Data read from memory: %d\n",*str);
    num=*str;
    *str='r';
    i=1;
    while(1){
        sleep(1);
        if(*str=='*'){
            *str=num*i;
            i++;
        }
        if(i>11){
            *str='b';
            break;
        }
    }
    shmdt(str);
    shmctl(shmid,IPC_RMID,NULL);
}
```

K191048  
Amman Soomro

```
return;  
}
```



```
ammansoomro@Amman-PC: ~/Visual_Studio/Lab 6  
ammansoomro@Amman-PC:~/Visual_Studio/Lab 6$ ./Q3_A  
Enter the number(1-10):10  
Data written in memory: 10  
Data written in memory: r  
10  
20  
30  
40  
50  
60  
70  
80  
90  
100  
ammansoomro@Amman-PC:~/Visual_Studio/Lab 6$  
ammansoomro@Amman-PC:~/Visual_Studio/Lab 6$ ./Q3_B  
Data read from memory: 10  
ammansoomro@Amman-PC:~/Visual_Studio/Lab 6$
```

## Task 4

### File A

```
#include<sys/types.h>  
#include<sys/ipc.h>  
#include<sys/shm.h>  
#include<stdio.h>  
#include<stdlib.h>  
#include<string.h>  
  
void main(){  
  
    int num;  
    key_t key = 1111;  
    int shmid = shmget(key,1024,0666|IPC_CREAT);  
    int *str = shmat(shmid,(void*)0,0);  
    *str=0;  
    while(*str!='b');  
    shmdt(str);  
    shmctl(shmid,IPC_RMID,NULL);  
    return;  
}
```

## File B

```
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<stdio.h>
#include<stdlib.h>
#include<string.h>

void main(){

    int num;
    key_t key = 2222;
    int shmid = shmget(key,1024,0666|IPC_CREAT);
    int *str = shmat(shmid,(void*)0,0);
    *str=1;
    while(*str!='b');
    shmdt(str);
    shmctl(shmid,IPC_RMID,NULL);
    return;
}
```

## File C

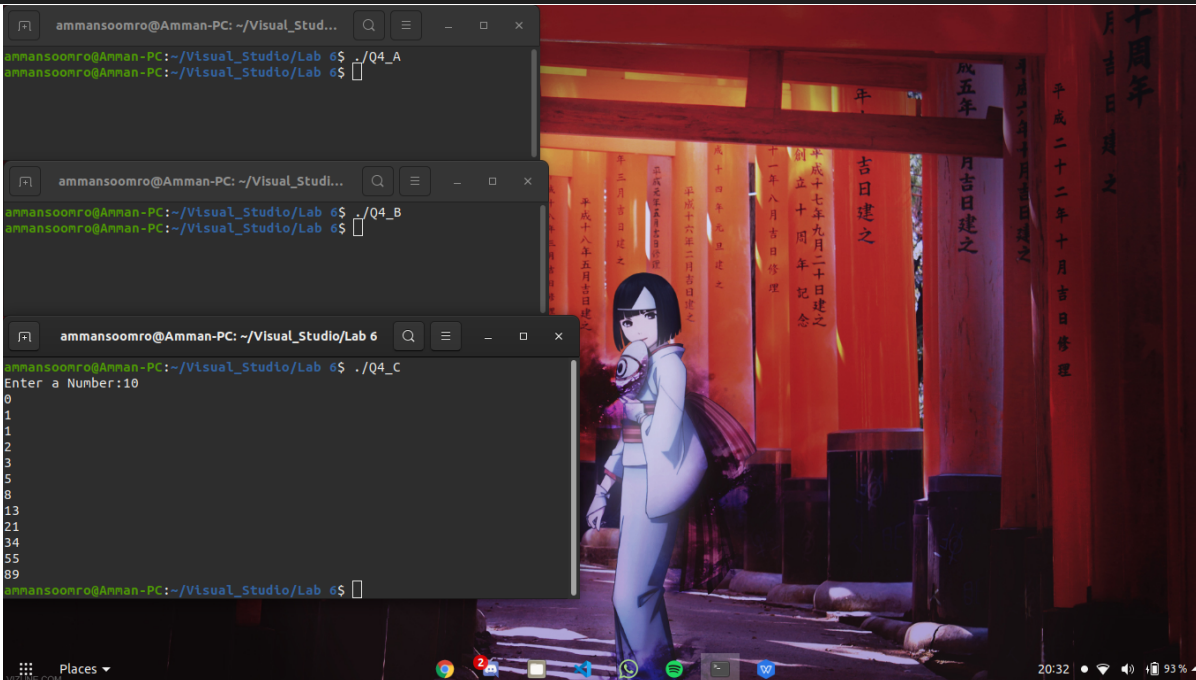
```
#include<sys/types.h>
#include<sys/ipc.h>
#include<sys/shm.h>
#include<stdio.h>
#include<stdlib.h>
void main(){

    int A,B,n;
    key_t key1 = 1111;
    key_t key2 = 2222;
    int shmid1 = shmget(key1,1024,0666|IPC_CREAT);
    int shmid2 = shmget(key2,1024,0666|IPC_CREAT);
    int *str1 = shmat(shmid1,(void*)0,0);
    int *str2 = shmat(shmid2,(void*)0,0);
    printf("Enter a Number:");
    scanf("%d",&n);
    printf("%d \n",*str1);
    printf("%d \n",*str2);
    while(n){
        A=*str1;
```



K191048  
Amman Soomro

```
B=*str2;
printf("%d \n",A+B);
*str1=*str2;
*str2=A+B;
n--;
}
*str1='b';
*str2='b';
shmdt(str1);
shmdt(str2);
shmctl(shmid1,IPC_RMID,NULL);
shmctl(shmid2,IPC_RMID,NULL);
return;
}
```



The screenshot shows a Windows desktop environment. The background is a digital illustration of a shrine with many wooden torii gates and vertical wooden pillars, some with Japanese text. A character with black hair and a white outfit is standing in the foreground. Overlaid on the desktop are three terminal windows. The top window has the title 'ammansoomro@Amman-PC: ~/Visual\_Stud...' and shows the execution of './Q4\_A'. The middle window has the title 'ammansoomro@Amman-PC: ~/Visual\_Stud...' and shows the execution of './Q4\_B'. The bottom window has the title 'ammansoomro@Amman-PC: ~/Visual\_Studio/Lab 6' and shows the execution of './Q4\_C', with the prompt 'Enter a Number:10' and a list of numbers: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89. The taskbar at the bottom shows various icons including a clock, network, volume, and battery status, along with the time '20:32' and '93%' battery level.