

11. A. Write a C/C++ program to illustrate the race condition.

```
#include<stdio.h>
#include<unistd.h>
#include<stdlib.h>
static void charatatime (char *);

int main (void)
{
    pid_t pid;
    if ((pid=fork( ))< 0)
    {
        printf("fork error\n");
    }
    else if(pid==0)
    {
        charatatime("Output from child\n");
    }
    else
    {
        charatatime("Output from parent\n");
    }
    exit(0);
}

static void charatatime(char *str)
{
    char *ptr;
    int c;
```

```

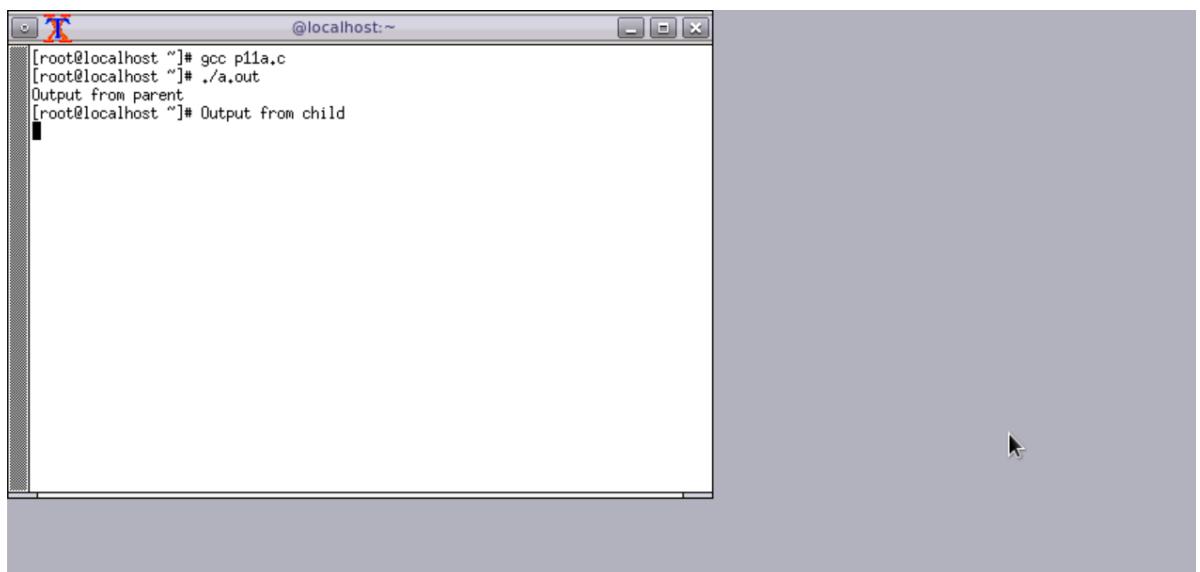
setbuf(stdout,NULL);      /* set unbuffered*/
for(ptr=str;(c=*ptr++)!=0;)
    putc(c,stdout);
}

```

OUTPUT:

Command for compiling : gcc pgmname.c

Command to execute : ./a.out



B) Write a C/C++ program which demonstrates inter-process communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.

```

#include<stdio.h>
#include<stdlib.h>
#include<fcntl.h>
#include<unistd.h>
#include<sys/types.h>
#include<string.h>

int main(int argc,char *argv[])
{
    int fd,num1,num2;
    char buf[100];

```

```

if(argc==3)
{
    mkfifo(argv[1],0666);
    fd=open(argv[1],O_WRONLY);
    num1=write(fd,argv[2],strlen(argv[2]));
    printf("no of bytes written = %d\n",num1);
}

if(argc==2)
{
    fd=open(argv[1],O_RDONLY);
    num2=read(fd,buf,sizeof(buf));
    buf[num2]='\0';
    printf("Message size %d received \n message received is = %s",num2,buf);
}

close(fd);

unlink(argv[1]);

return 0;
}

```

Program Output:

Terminal 1

```

[root@localhost ~]# vi p11b.c
[root@localhost ~]# cc p11b.c -o writerprocess
[root@localhost ~]# writerprocess pipe1 helloworld
Number of bytes written = 10

```

Terminal 2

```
[root@localhost ~]# vi p11b.c
```

```
[root@localhost ~]# cc p11b.c -o readerprocess
```

```
[root@localhost ~]# readerprocess pipe1
```

Message of size 10 bytes received

Message received = helloworld

The image shows two terminal windows side-by-side. Both windows have a red 'X' icon in the top-left corner and a title bar '@localhost:~'. The left window contains the following text:

```
[root@localhost ~]# gcc p11b.c
p11b.c: In function 'main':
p11b.c:13:1: warning: implicit declaration of function 'mkfifo' [-Wimplicit-function-declaration]
  13 | mkfifo(argv[1],0666);
     | ^~~~~~
[root@localhost ~]# ./a.out pipe1 helloworld
no of bytes written=10
[root@localhost ~]#
```

The right window contains the following text:

```
[root@localhost ~]# gcc p11b.c
p11b.c: In function 'main':
p11b.c:13:1: warning: implicit declaration of function 'mkfifo' [-Wimplicit-function-declaration]
  13 | mkfifo(argv[1],0666);
     | ^~~~~~
[root@localhost ~]# ./a.out pipe1
the message size 10 bytes received
message received is= helloworld[root@localhost ~]#
```