OS Lab Experiment VII

Varun Khadayate

NMIMS

# AIM

To Implement Readers and Writers problem using:

Case-1: Semaphores

Case-2: Monitors

# CODE

## Semaphores

#include<iostream>

#include<conio.h>

#include<stdio.h>

#include<stdlib.h>

using namespace std;

int mutex=1, wrt=0, readcnt=0, writecnt=0;

char buffer[500];

wait(int \*s)

{

\*s-=1;

}

signal(int \*s)

{

\*s+=1;

}

void readerenter()

{

if(writecnt==0)

{

wait(&mutex);

readcnt++;

cout<<"Reader "<<readcnt<<":"<<endl;

cout<<buffer<<endl;

signal(&mutex);

}

else

cout<<"Write mode on please wait !!"<<endl;

}

void readerexit()

{

if(readcnt>0)

{

wait(&mutex);

readcnt--;

signal(&mutex);

}

else if(readcnt==0)

cout<<"No readers left"<<endl;

else

cout<<"Write mode on please wait"<<endl;

}

void writerenter()

{

if(writecnt==1)

cout<<"Only one writer allowed"<<endl;

else if(readcnt==0&&writecnt==0)

{

wait(&mutex);

wait(&wrt);

++writecnt;

cout<<"Input:"<<endl;

gets(buffer);

signal(&wrt);

signal(&mutex);

}

else if(readcnt>0)

cout<<"Process still has readers please wait !!"<<endl;

else if(writecnt>0)

cout<<"There is a writer in process. Please Wait !!"<<endl;

}

void writeexit()

{

if(writecnt>0)

{

wait(&mutex);

wait(&wrt);

--writecnt;

signal(&wrt);

signal(&mutex);

}

else if(writecnt==0)

cout<<"No writers here"<<endl;

else

cout<<"System not inwrite mode"<<endl;

}

void status()

{

cout<<"Buffer contents:"<<endl;

cout<<buffer<<endl<<endl;

if(readcnt>0)

cout<<"System is in read mode with "<<readcnt<<" readers"<<endl;

else if(writecnt>0)

cout<<"System is in write mode with "<<writecnt<<" writers"<<endl;

else

cout<<"System is in neutral mode with no readers or writers"<<endl;

}

int main()

{

system("cls");

int m;

CHOICES:cout<<"\t\t"<<"MENU"<<endl;

cout<<"1. Enter read mode"<<endl;

cout<<"2. Exit read mode"<<endl;

cout<<"3. Enter write mode"<<endl;

cout<<"4. Exit write mode"<<endl;

cout<<"5. Status"<<endl;

cout<<"6. Exit"<<endl;

cout<<"Choice:";

cin>>m;

system("cls");

switch(m)

{

case 1: readerenter();

break;

case 2: readerexit();

break;

case 3: writerenter();

break;

case 4: writeexit();

break;

case 5: status();

break;

case 6: goto CONT;

}

goto CHOICES;

CONT:

return 0;

}

## Monitors

#include<iostream.h>

#include<conio.h>

#include<stdio.h>

#include<stdlib.h>

int no\_read = 0, no\_write = 0, read\_track = 0, write\_track = 0;

char buffer[500];

wait(int \*s)

{

\*s-=1;

}

signal(int \*s)

{

\*s+=1;

}

class monitor

{

public:

void readerenter()

{

if(no\_read > 0 || no\_read == 0 && no\_write == 0)

{

++no\_read;

wait(&read\_track);

cout<<"Reader "<<no\_read<<":"<<endl;

cout<<buffer<<endl;

signal(&read\_track);

}

else if(no\_write>0)

cout<<"Writing in progress please wait"<<endl;

}

void readerexit()

{

if(no\_read > 0)

--no\_read;

if(no\_read==0)

cout<<"No read in progress"<<endl;

}

void writerenter()

{

if(no\_read>0)

cout<<"Reading in progress. Writers not allowed"<<endl;

else if(no\_write == 0 && no\_read == 0)

{

++no\_write;

wait(&write\_track);

cout<<"Input:"<<endl;

gets(buffer);

signal(&write\_track);

}

else if(no\_write == 1)

cout<<"Only one writer allowed"<<endl;

}

void writerexit()

{

if(no\_write == 1)

--no\_write;

if(no\_write == 0)

cout<<"No write in progress"<<endl;

}

}mon;

void status()

{

cout<<"Buffer contents:"<<endl;

cout<<buffer<<endl<<endl;

if(no\_read>0)

cout<<"System is in read mode with "<<no\_read<<" readers"<<endl;

else if(no\_write>0)

cout<<"System is in write mode with "<<no\_write<<" writers"<<endl;

else

cout<<"System is in neutral mode with no readers or writers"<<endl;

}

void main()

{

clrscr();

int m;

CHOICES:cout<<endl<<"Enter your choice:"<<endl;

cout<<"1. Enter read mode"<<endl;

cout<<"2. Exit read mode"<<endl;

cout<<"3. Enter write mode"<<endl;

cout<<"4. Exit write mode"<<endl;

cout<<"5. Status"<<endl;

cout<<"6. Exit"<<endl;

cin>>m;

clrscr();

switch(m)

{

case 1: mon.readerenter();

break;

case 2: mon.readerexit();

break;

case 3: mon.writerenter();

break;

case 4: mon.writerexit();

break;

case 5: status();

break;

case 6: goto CONT;

}

goto CHOICES;

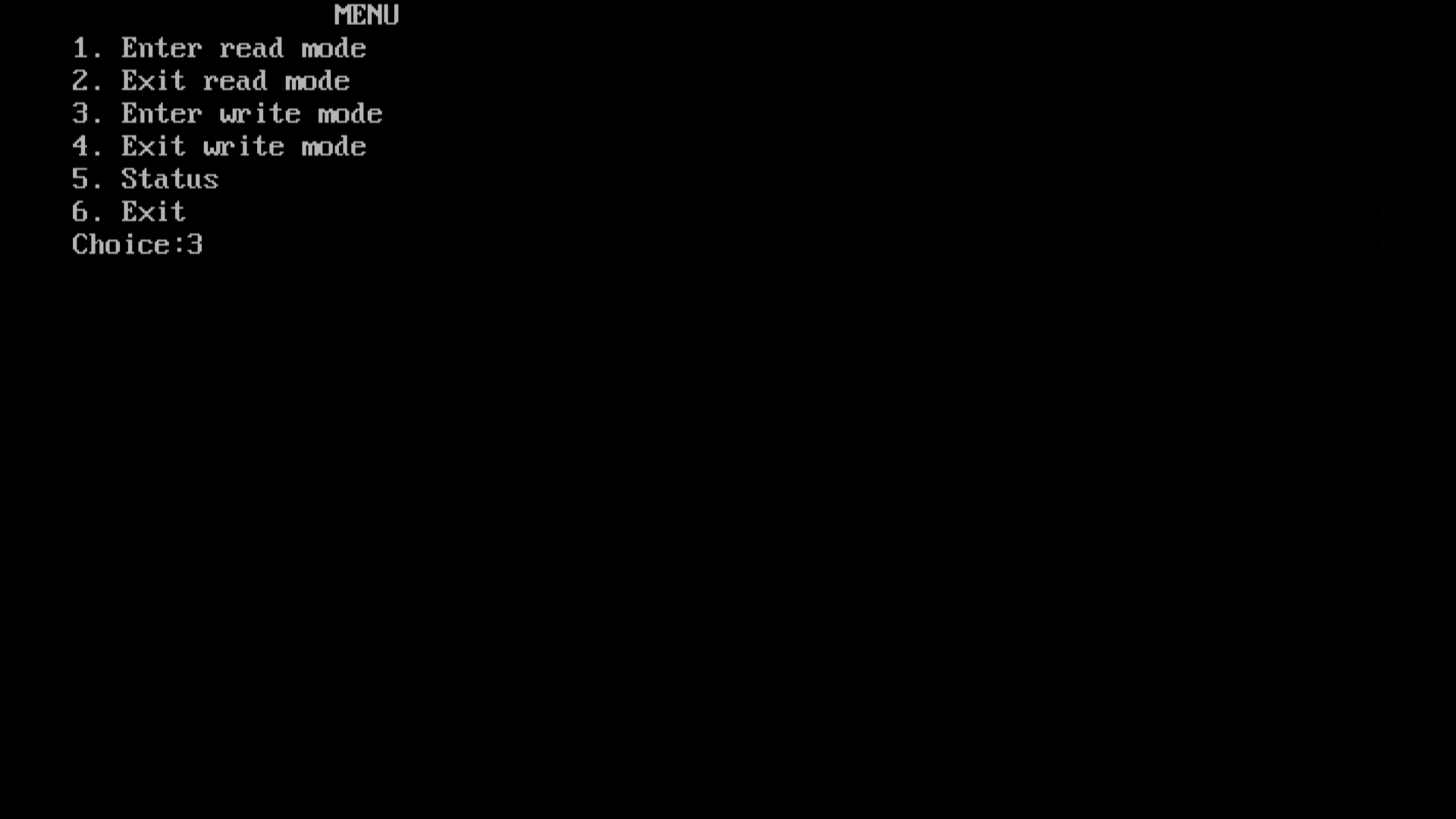
CONT:

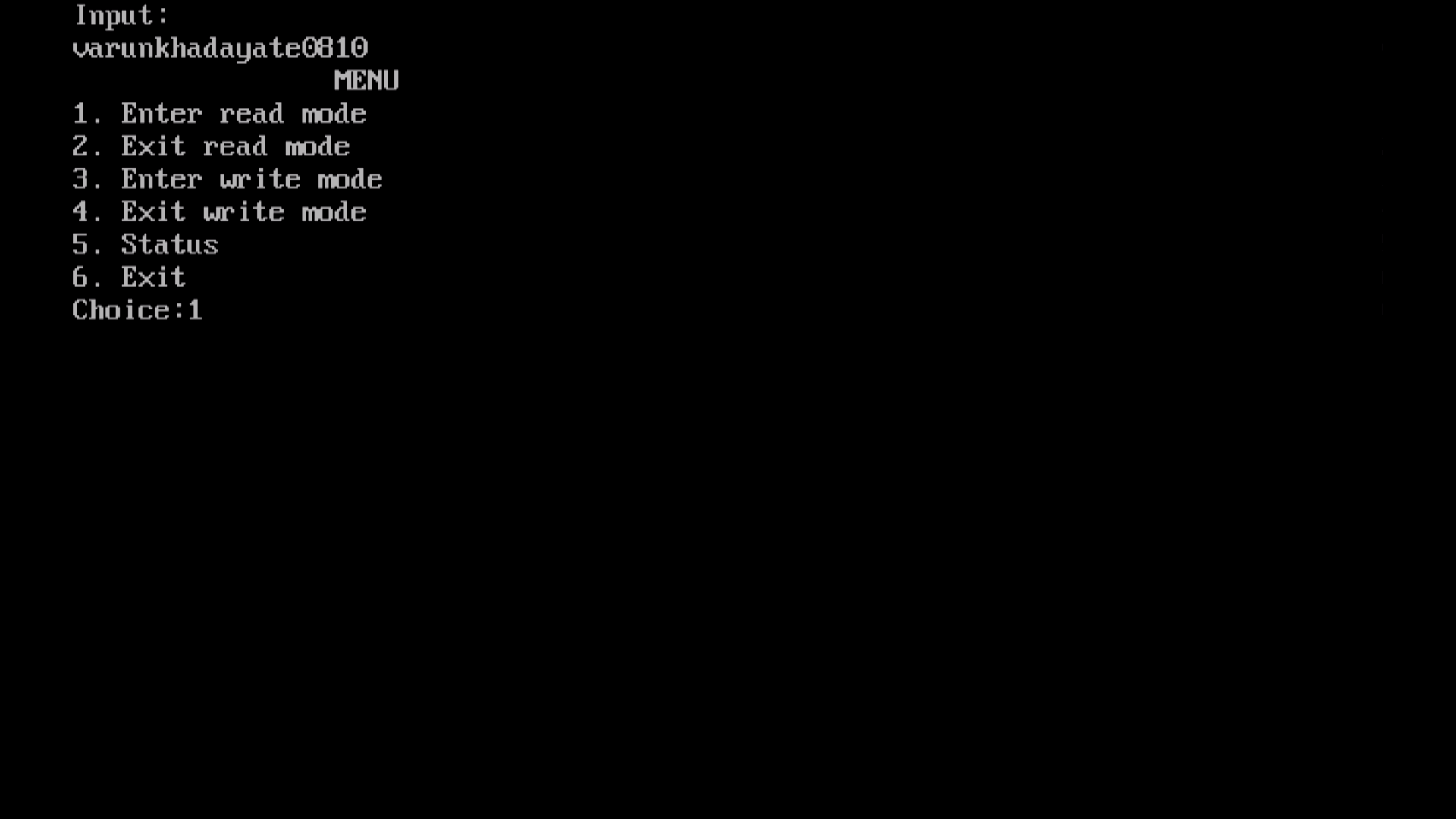
getch();

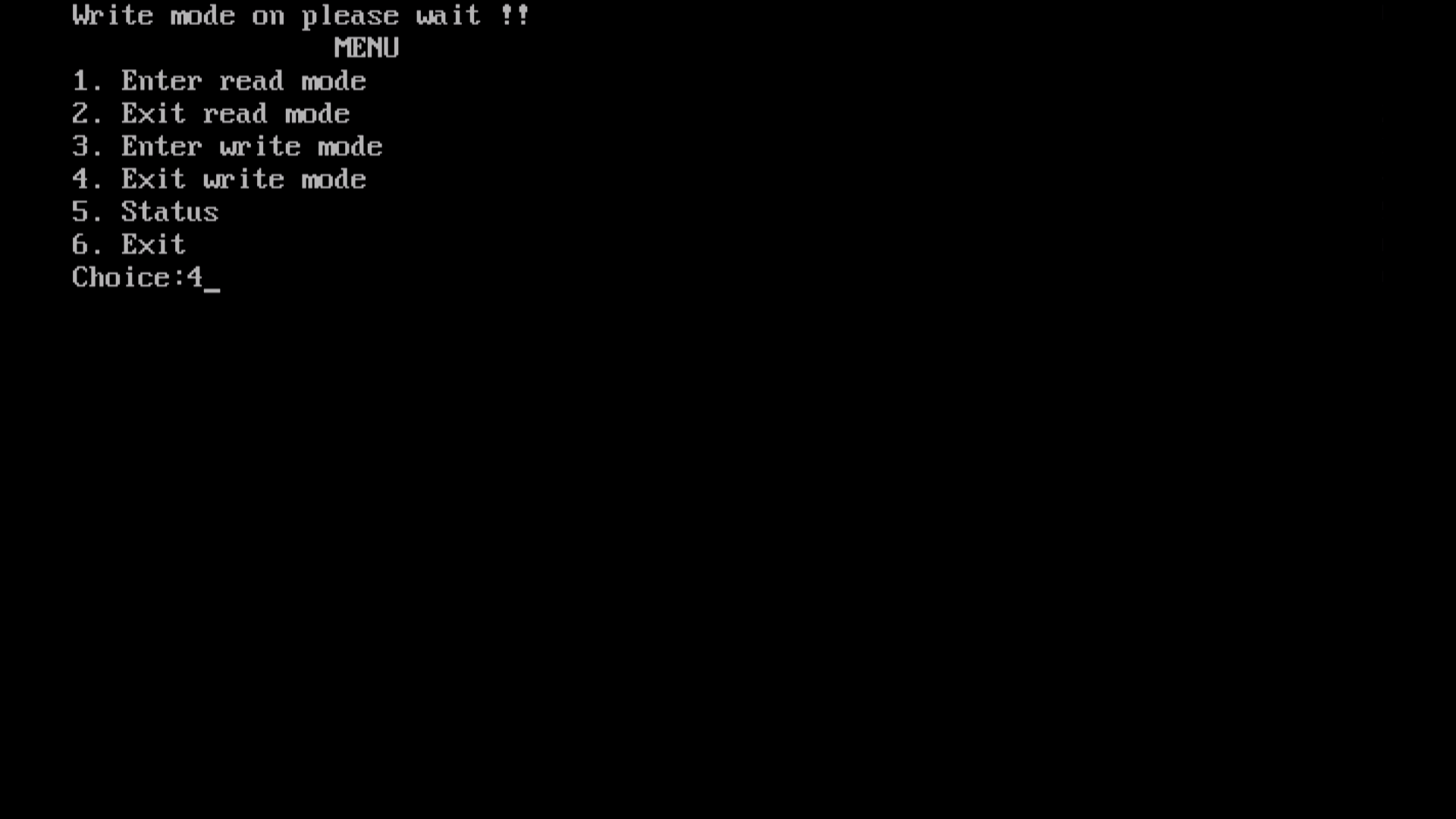
}

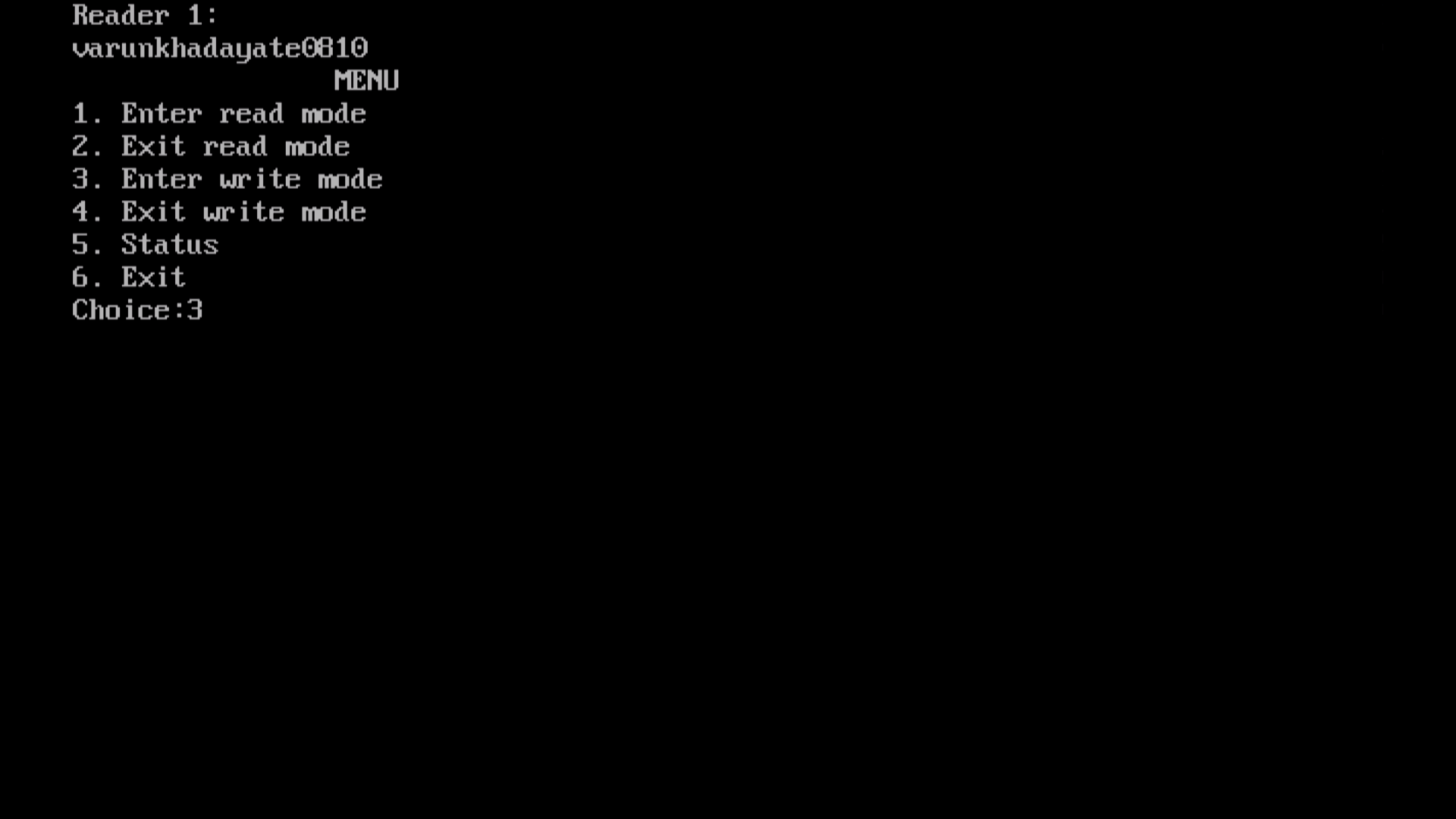
# Code Snapshots

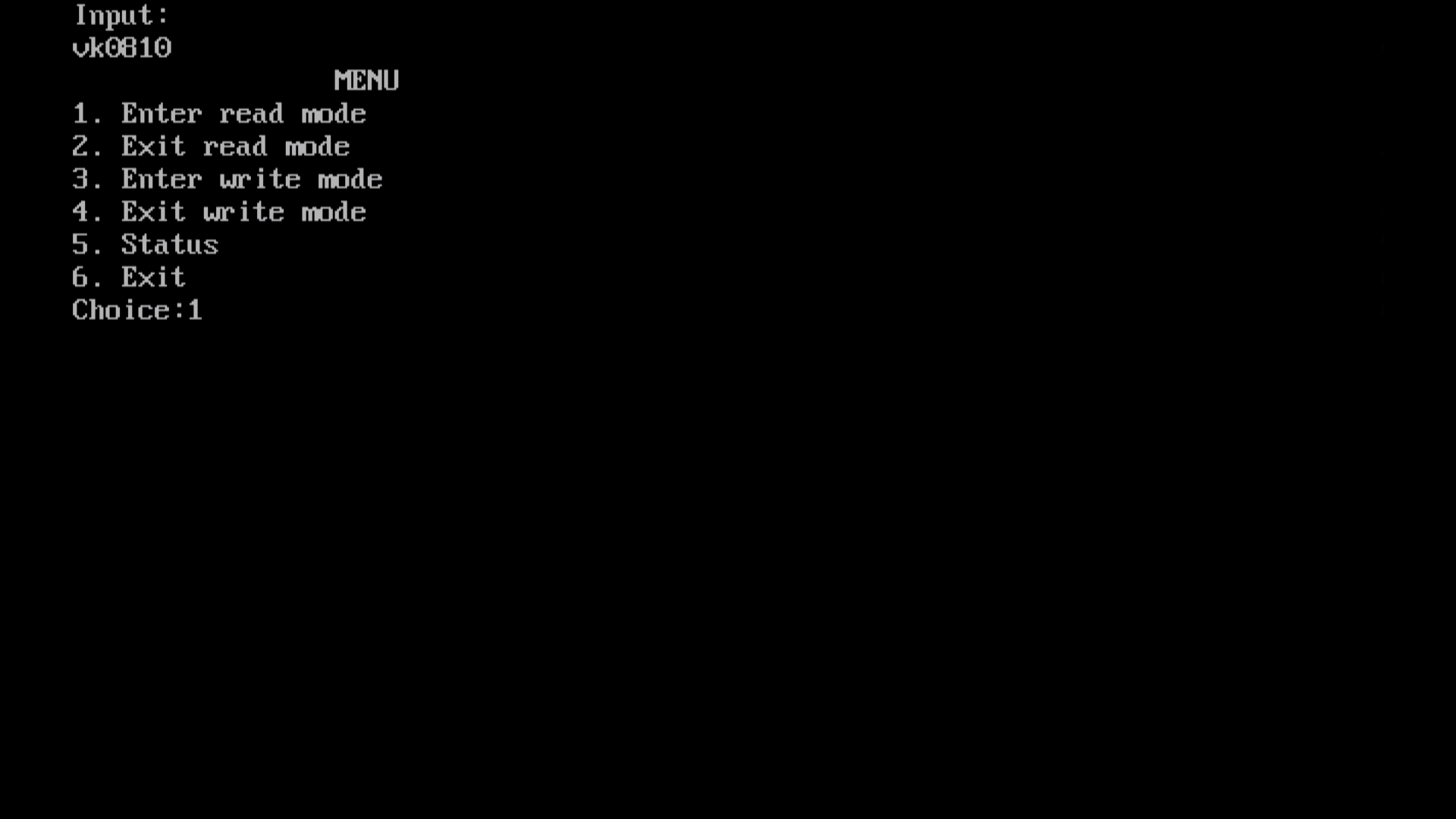
## Semaphores

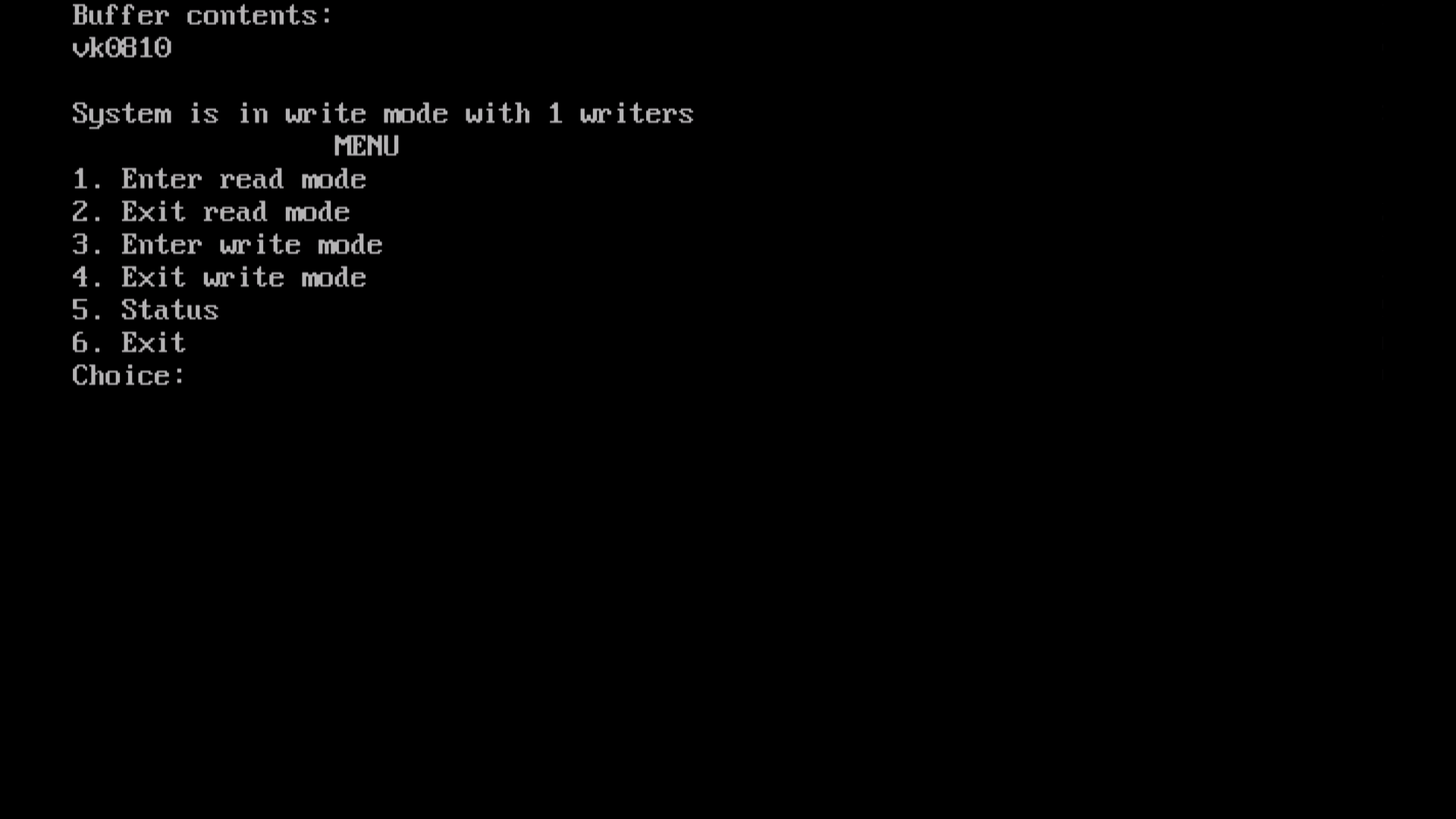












## Monitors

