Rešenja trećeg kolokvijuma iz Operativnih sistema 2, Januar 2013.

1. (10 poena)

```
class Mutex {
public:
  Mutex ()
    { InitializeCriticalSectionAndSpinCount(&criticalSection,0x00000400); }
 ~Mutex()
    { DeleteCriticalSection(&criticalSection); }
  void enter ()
    { EnterCriticalSection(&criticalSection); }
  void exit ()
    { LeaveCriticalSection(&criticalSection); }
private:
  CRITICAL_SECTION criticalSection;
};
```

2. (10 poena)

```
#!/bin/bash

if [ $# -lt 2 ];then
    echo "Nedovoljan broj argumenata!"
    exit 1
fi

tmp="tmp.html"
wget "$1" -O $tmp
if [ $? -ne 0 ];then
    echo "Nepostojeci URL"
    exit 2
fi

IFS_old=$IFS
IFS=$'\n'
for i in $(cat $tmp | grep href=\".*\.$2\"\>| sed
's/.*href="\(.*\)">.*/\1/');do
    echo "$i"
    wget "$i"
done
IFS=$IFS_old
rm $tmp
```

3. (10 poena)

```
class Agent {
public:
 Agent(key_t key);
 virtual ~Agent();
 void takeTobaccoAndPaper () {atomicOnTwoSems(Paper,Tobacco,-1);}
 void takePaperAndMatch   () {atomicOnTwoSems(Paper,Match,-1);}
 void takeTobaccoAndMatch () {atomicOnTwoSems(Match,Tobacco,-1);}
 void finishedSmoking () {atomicOnTwoSems(randNum(),randNum(),1);}
private:
 int id;
 void atomicOnTwoSems(int first, int second, int op);
 int randNum();
 static const int Paper=0, Match=1, Tobacco=2;
};
```

```
void Agent::atomicOnTwoSems(int first, int second, int op){
  struct sembuf sems[2];
     sems[0].sem_num = first;
     sems[1].sem_num = second;
     sems[0].sem_op = sems[1].sem_op = op;
     sems[0].sem_flg =  sems[1].sem_flg = SEM_UNDO;
     semop(id, sems, 2);
 }

int Agent::randNum() {
 static int prev;
 int next = rand()%3;
 prev =(next==prev)?++prev%3:next;
 return  prev;
}

Agent::Agent(key_t key) {
  id = semget(key, 3, 0666 | IPC_CREAT);
  for (int var = 0; var < 3; ++var)
       semctl(id, var, SETVAL, 0);
  finishedSmoking();
}

Agent::~Agent() {
  for (int var = 0; var < 3; ++var) {
   semctl(id, 0, IPC_RMID);
  }
}
```