socket(AF\_INET, SOCK\_STREAM, IPPROTO\_TCP);

bind(s\_socket, (struct sockaddr \*)&s\_addr, sizeof(s\_addr)

listen(s\_socket, 5)

accept(s\_socket, (struct sockaddr \*)&c\_addr, sizeof(c\_addr);

write(c\_socket, buffer, strlen(buffer));

htol(INADDR\_ANY), htons(PORT)

inet\_addr(IPADDR), inet\_ntoa(c\_addr.sin\_addr));

public class BeepTask implements Runnable { }

Runnable beepTask = new BeepTask();

Thread thread = new Thread(BeepTask);

Thread thread = new Thread(new Runnable() { });

public calss BeepThread extends Thread { }

Thread thread = new BeepThread();

Thread thread = new Thread() { };

thread.getName, thread.setName,

Thread.currentThread();

thread.setPriority(Thread.MAX\_PRIORITY);

public synchronized void setMemory(int memory) { }

public synchronized void openDoor(String name) { }

serverSocket = new ServerSocket(5001);

serverSocket = new ServerSocket();

InetSocketAddress ipep = new InetSocketAddress(5001);

serverSocket.bind(ipep);

Socket socket = serverSocket.accept();

InetSocketAddress isa

= (InetSocketAddress) socket.getRemoteSocketAddress();

isa.getAddress().getHostAddress()

InputStream is = socket.getInputStream();

OutputStream os = socket.getOutputStream();

BufferedReader in

= new BufferedReader(new InputStreamReader(is));

PrintWriter out = new PrintWriter(os, true);

msg.equals(“bye”)