**SCHOOL OF COMPUTING, ENGINEERING AND BUILT ENVIRONMENT**

**Department of Computing**

**BSc/BSc (Hons) Software Development for Business**

**BSc/BSc (Hons) Computing**

**Systems Programming**

**Module code M3I324183-18-B**

Testing Document

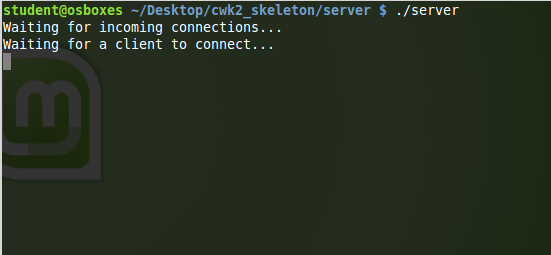
|  |  |  |
| --- | --- | --- |
| **Michael Kofi Badu** | **S1719029** | **mkofib200@caledonian.ac.uk** |

TEST DOCUMENT FOR A CLIENT SERVER APPLICATION USING SOCKET MECHANISM WRITTEN IN C LANGUAGE

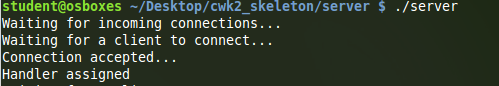
In order to commence the test, the first thing is to **establish the client and server connection**

Server setup

The server sets up the connection waiting to connect to the client server

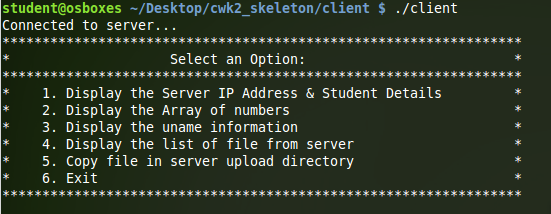


When the client is connected, the server displays the figure below, indicating that the client has been successful connected.



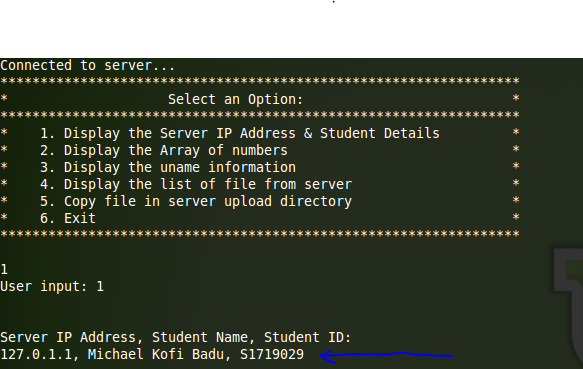
Client Setup

Ones the connection is established the menu is then displayed to the client and a notification informing the user of the system “Connected to server...”

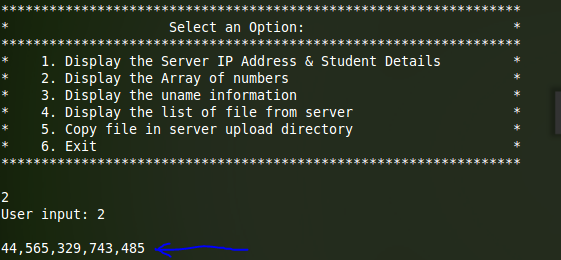


|  |  |  |  |
| --- | --- | --- | --- |
| OPTION | TEST | RESULTS EXPECTED | ACTUAL RESULTS DISPLAYED |
| 1 | Display the server IP Address and Student Details | When this option is selected, it is expected that the system displays the server’s IP address, the student name and student ID Number | Upon selection of the option the system displays, the server’s IP address, the student name and the student number respectively.  NB: *Test results illustrated as figure 1 below.* |
| 2 | Display the array of numbers | This option is expected to show an array of random 5 numbers from 0 to 1000. | The system displays a list of random 5 numbers from 0 to 1000. Showing that 5 random numbers have been generated from the server in an array to the client.  NB: *Test results illustrated as figure 2 and 2.1 below.* |
| 3 | Display uname information | Displays information on hostname, system name, details of the version and machine. | Actually displays information on hostname, system name, details of the version and machine.  NB: *Test results illustrated as figure 3* |
| 4 | Display the list of file from the server | The function is expected to list out all the files existing in the directory | The system outlines all the existing files in the directory and then displays them for the user  NB: *Test results illustrated as figure 3* |
| 5 | Retrieving or copying files from the server‘s ‘upload‘ directory | It is expected that the function hints the user to enter the file name they would want to retrieve from the ‘upload‘ directory  after it has been deleted from the folder.  Then after the file should be found back in its initial directory with its contents. | The system hints the user to enter the file name they require to retrieve or copy after it has been deleted from the folder  NB: *As illustrated in figure 5 which makes the user specify the file.*  Then after the file is found back in its initial directory with its contents.  NB: *Test results illustrated as figure 5.1 when the file to be retrieved is specified* |
| 6 | Exit | The application is expected to stop and exit the menu notifying the user with a “Goodbye!” | Ones the input is chosen the application stops and exits the menu notifying the user with a “Goodbye!” |

**Figure 1:** Shows server IP address, name and ID

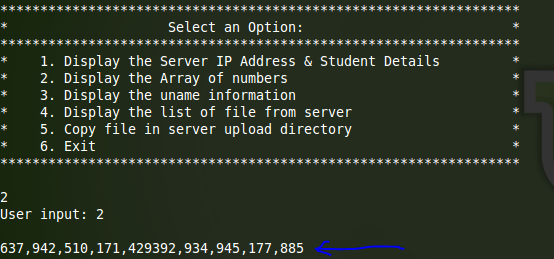
****

**Figure 2:** random numbers

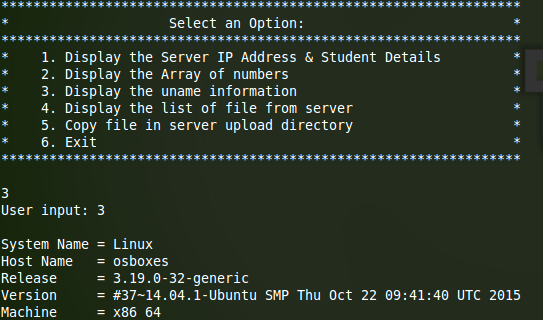
****

**Figure 2.1**

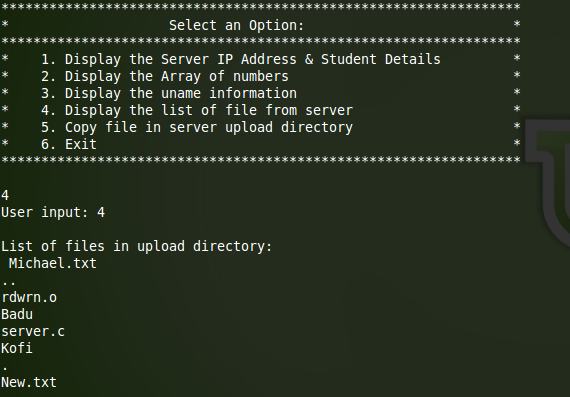
NB: When option 2 is re-entered again, displays an additional 5 random numbers.

****

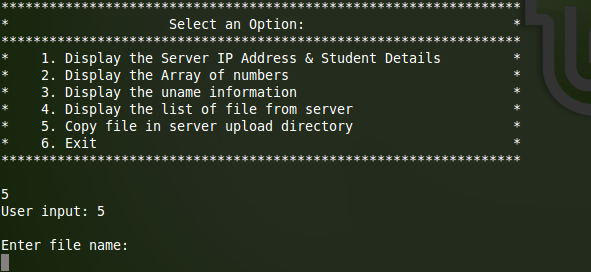
**Figure 3:** uname information

****

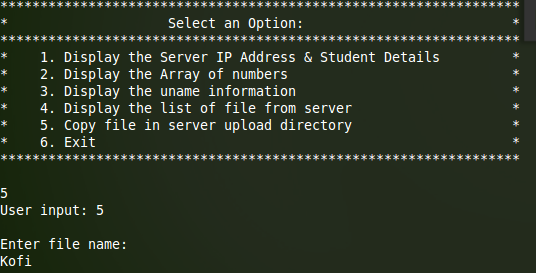
**Figure 4:** List of files

****

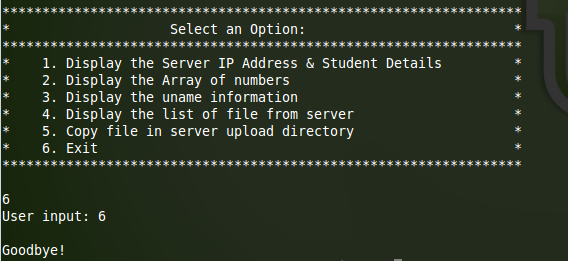
**Figure 5:** Enter file name you want to retrieve

****

**Figure 5.1:** Copy and retrieve the file(s)

****

**Figure 6:** Exit

****