

## Introductory information (19-09-2001)

We expect you already know how to “use” computer. Reading emails, writing text... etc.

If you do not have any experience, there is Computer Literacy course.

### Login to proffa or Birdland network:

As a student of TUT you have access to student UNIX account in computer called “proffa.cc.tut.fi”.

As a student in Department of Information Technology you have UNIX account in “Birdland” computer network (domain cs.tut.fi).

Both requires secure login.

1. Use **ssh client** (should be at desktop).
2. Set computer name (**proffa.cc.tut.fi** or **kaarne.cs.tut.fi**)
3. Use your login
4. Enter password

### After you are logged into UNIX server:

Your commands are executed on “remote” machine – server.

On page <http://www.cs.tut.fi/lintula/eng/> you can find tutorials how to use UNIX. Or to learn about commands, use **man** command.

Example commands:

```
cd directory - change directory
pwd          - prints current directory
more file   - show content of file
pico file   - file editor
joe file    - another file editor
g++ file    - C++ compiler
ld          - linker
rm file     - remove (delete) file
mv file1 file2 - change name file1 to file2
cp file1 file2 - copy file1 to file2
mkdir directory - create new directory
ls          - prints content of directory
```

For example:

```
man cp
will print you information on how to use cp command
```

### **First program:**

For historical reasons (first program in C language) first programs does not do much. Just prints “Hello World!” and exits.

Example:

Our program will be in C++. Therefore we will create file with source code in C++, compile it and run to see if it works.

Let say we will have file with name hello.cc.

Contents of file will be:

```
/*
 * First program in C++.
 * /
#include <iostream.h>

void main()
{
    cout << “Hello World!” << endl;
}
```

Once we have this program in file, we can compile it.

```
% g++ hello.cc -o hello
```

Note that `-o hello` will give compiler order to create executable file with name hello. Otherwise compiler create executable with name a.out.

Once program is successfully compiled we can run it.

```
% ./hello
```

In case of error in your program source, compiler will print possible reason for error with line number where error seems to happen.

### **Task:**

Modify program to read your name from keyboard and print “Welcome your\_name”.

To store your name, use variable of character type.

Hint:

```
char buffer[50];
create variable of character type that can hold up to 50 characters
```

```
cout << "Hi there! " << "How is " << buffer << "?"<<endl;
```

shows how to concatenate output text from number of strings.

**Own work:** If you like to try programs from lectures, just go on.