Lab 5: A Third C Program

Aim
This lab class gives you an opportunity to practise new C programming skills and in particular:

- work with input and output;
- work with pointer variables; and
- refresh your knowledge of creating Projects in Visual Studio.

Context
Variables can be thought of as named memory locations. Crudely each is just a box that values can be placed into and read from.

When we want to display values, we use the `printf()` function, specify the kind of value we want to see (using a format string), and give the value we want to print. This value is often just a variable name.

When we want to get a value from the user we need to use the `scanf()` function, specify what kind of value we want (using a format string), and the box we want to put the value into. This box is either the address of a variable or a pointer to the variable that we want to change. It cannot be the name of the variable itself.

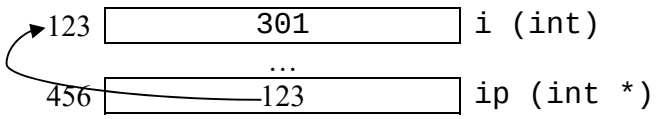
Tasks
1. Create a new **Microsoft Visual Studio** C project with name `Lab5_1` (see instructions from last week if necessary). Within the project, create a C source file called `variables.c`.

   Place the content of `variables.c` from MyLO (shown below) into your file, save it, compile and run it (removing any errors you may find) by selecting "Start without debugging" from the **Debug** menu (or pressing the <*Control*>-<*F5*> key combination).

   ```
   #include <stdio.h>

   int main(int argc, char *argv[])
   {
       double d;

       printf("Please enter a floating point number: ");
       scanf("%lf",&d);

       printf("The number you entered was: %lf\n",d);
   }
   ```

   Look at the code and modify it so that it asks the user for an integer and outputs not only the integer but also the address of the integer. Note: the format string for a `double` (long float) is `%lf`, the format for an `int` is `%d`, and the format for an address is `%p`.

2. Add a pointer to an integer (`int *`) variable to the program, set it to the address of the `int` variable, and then use the pointer variable for the remainder of the program rather than the `int` variable. A diagram of the situation (showing the `int i` at address 123 and value 301 and the pointer, `ip`, at address 456) is shown below.



Display the value of the pointer variable and the address of the `int` variable.

3. Create a new **Microsoft Visual Studio** C project with name `Lab5_2`. Within the project, create a C source file called `max.c`.

   Write a C program to fill an array of 10 elements with `int` values that the user provides and then find and display the largest value.

   You should define a constant for the size of the array; if you have difficulty using a constant, `#define` a symbol instead.

4. Work on your assignment! Read it, ask questions, understand it, and try to complete one of the functions.