Including files

- **Including files**
  - The `#include` directive:

```
#include  <filename>      - include file "filename" from
                             the C's standard include directory
                             (which is: /usr/include)

#include "filename"       - include file "filename" from
                             a user specified diretory
```

The **default user specified include directory** is:

- The **current directory**

  - Specifying **user specified include directories**:

```
gcc  -Idir1  -Idir2 ....  C-program
```

The **C compiler** will look for **include files** included with `#include "..."` in directories `dir1`, `dir2`, etc.

  - **Common practice:**

- **Included file** in **C programs** ***usually*** ends with the `.h` extention

  `.h` = **header file extension.**

  - The **include files** are called ***header* files** because the `#include` command ***almost* always** is located at the **top (head)** of a **C program file** !!!

  - **Example:**

file1.h	include1.c

`#define square(x)  (x*x)`	`#include "file1.h"`
	
	`int main( int argc, char* argv[] )`
	`{`
	`  double a, b;`
	
	`  b = square(a);`
	`}`

Output of `gcc -E include1.c`:

```
int main( int argc, char* argv[] )
{
   double a, b;

   b = (a*a);
}
```

- **Note:** ***header*** **file**

  - The `#include` command is **often** at the **start (head)** of a **C program**

    Hence, the **included files** are called **header files** in **C**

- ***Nested*** **includes**

  - **Fact:**

    - The `#include` directives can **nest**

      I.e.: an **included file** can contain `#include` directives to **include** ***other*** **files**

  - **Example: header1-1.h** includes **another** header file **header1-2.h**:

    - **header1-1.h**:

```
#include "header1-2.h"

int x;         // define variable x
```

    - **header1-2.h**:

```
int y;        // define variabel y
```

    - The **main** program:

```
#include "header1-1.h"    // Will define variable x and y
```

```
int main(int argc, char * argv[] )
{
   x = 1;
   y = 2;

   printf("x = %d, y  %d\n", x, y);
}
```

- **How to do demo:**

```
cd  ~cs255000/demo/c/Cprep
gcc -E main1.c
```

- ***Recursive* includes**
  - **Warning:**

    - You can **create *recursive* file inclusion**

  - **Example: recurse.h**

```
#include "recurse.h"           // Include the header file itself...

int x;
```

Result:

```
recurse.h:2:21: error: #include nested too deeply
```

  - **How to do demo:**

```
cd  ~cs255000/demo/c/Cprep
gcc -E main-recurse.c > /dev/null
```