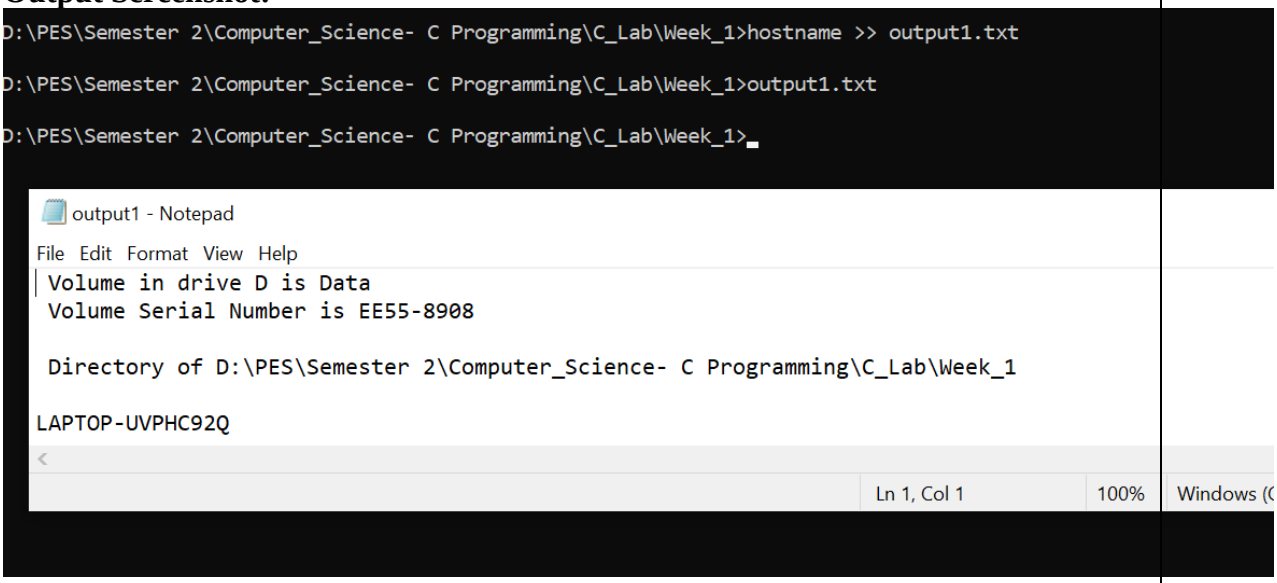
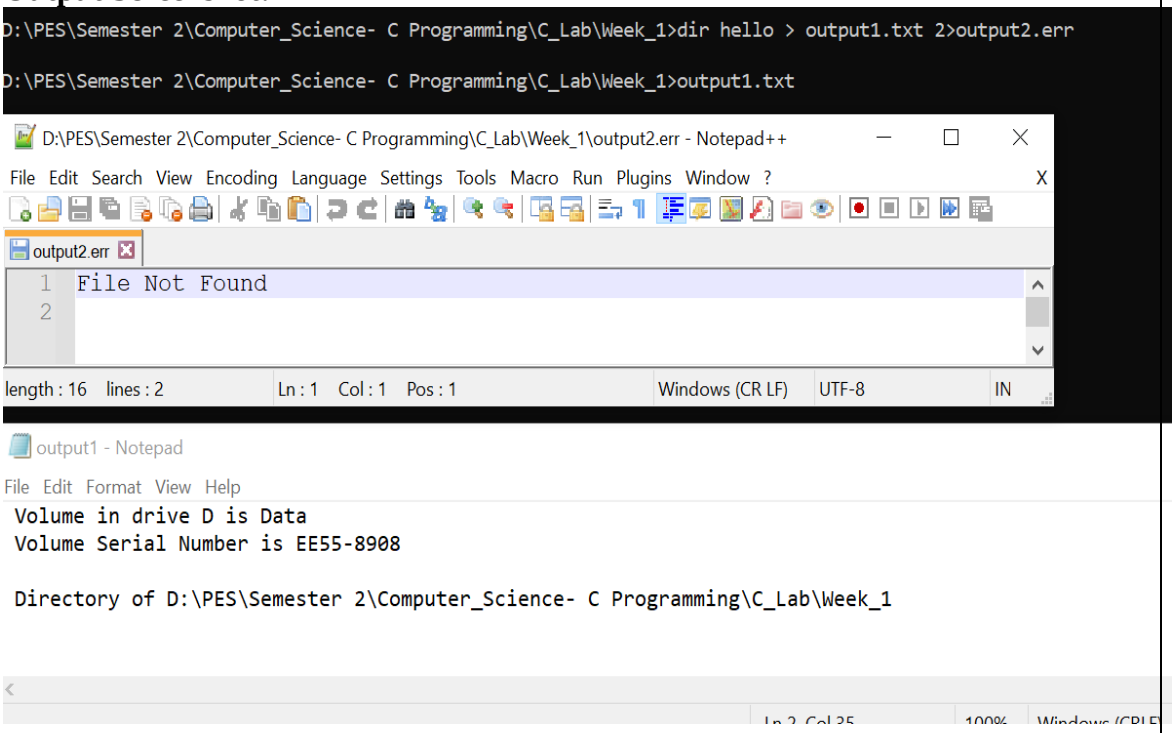
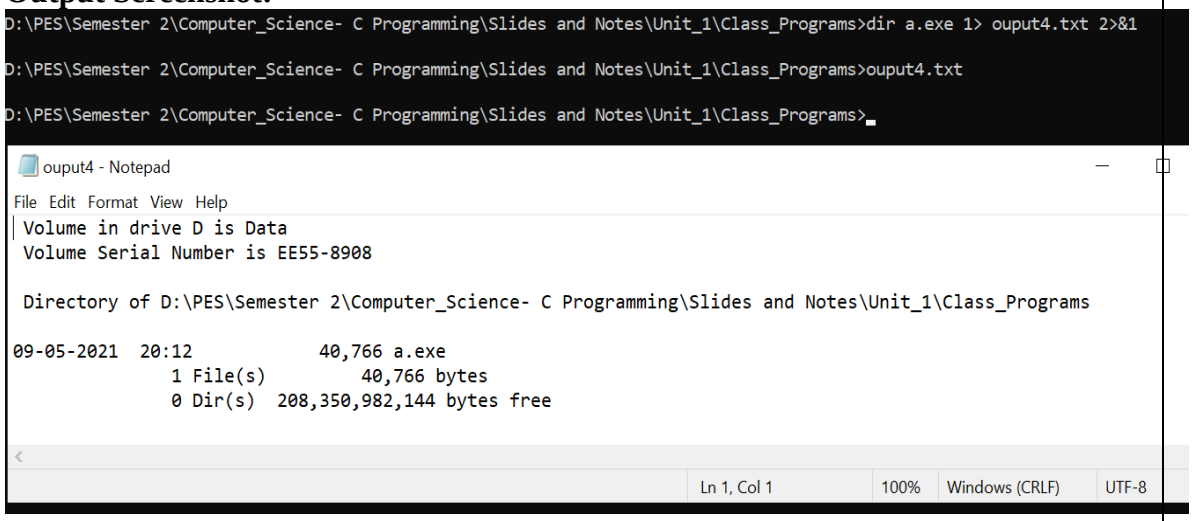
Week 1: File Redirection and gcc compiler options

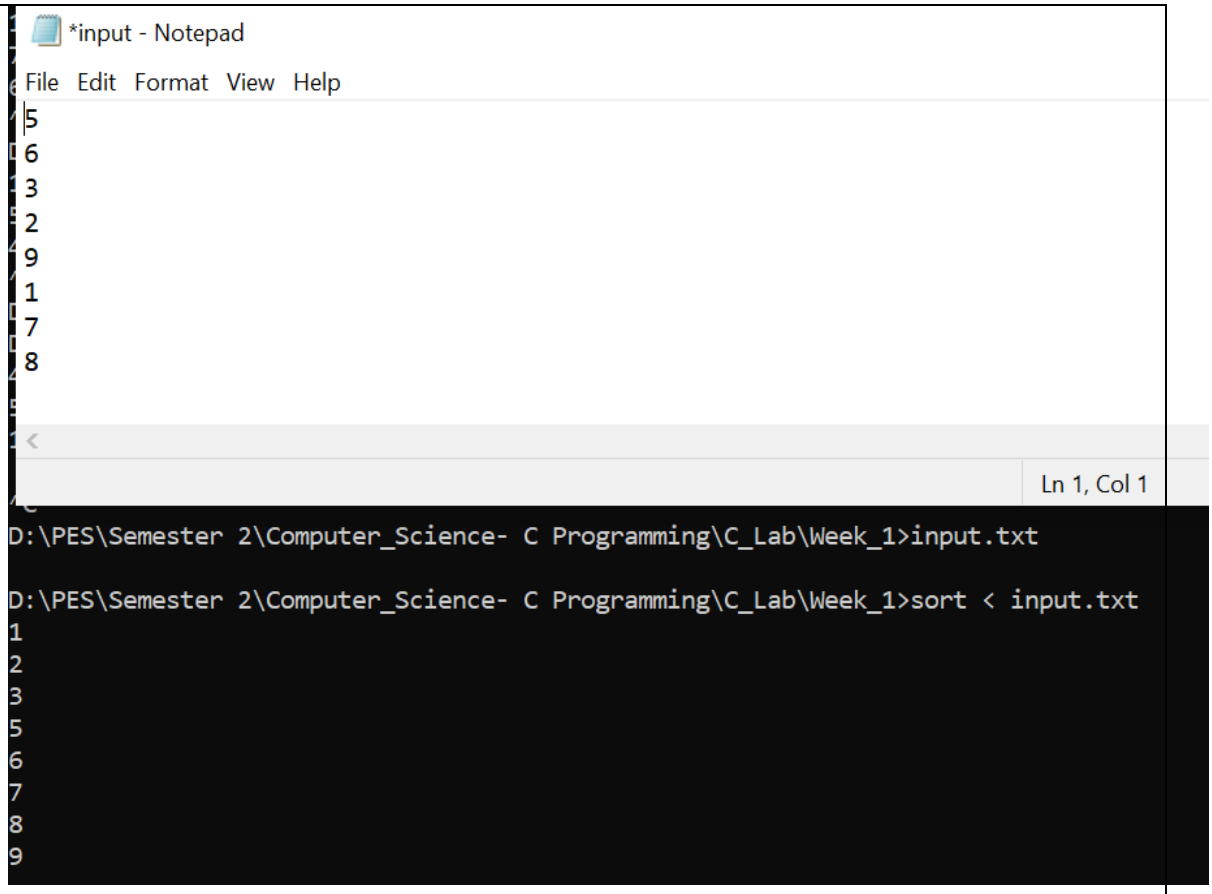
Name: Naren Chandrashekhar	SRN: PES2UG20CS216	Section: G
	Date: 06/05/2021	Week Number: 1

Task 1	**Redirect Standard Output :Write to New File**
	Description: To redirect standard output to a new file we have to ue the following command dir file_name > output_file_name For example dir 1.txt > output1.txt 1. dir command is used to display the folder contents and also checks whether a particular file is found or not 2. 1.txt is the file name we are going to check if present in the folder 3. The > character tells the console to output **STDOUT** to the file with the name we have provided. In the example given above output1.txt is the name of the file where the output will be redirected 4. We can view the standard output that went to the file by typing "output1.txt" in the command window. This will open the text file in your default text file viewer. For most of them, this is usually Notepad.exe.
	Commands: **dir file_name - Checks whether a file exits or not** **dir file_name > output_file_name -Redirects standard output to a new file**
	Output Screenshot: 

Task 2	**Redirect Standard Output: Writes to the Same File**
	Description: To redirect standard output to the same file we use >>. For example to redirect the output of command hostname to the file output1.txt which was already created in TASK1 the folloeing command can be used. hostname >> output1.txt It appends the existing file with new contents
	Commands: hostname >> file_name
	Output Screenshot: 
Task 3	**Redirect Standard Error To a File**
	Description: To redirect a standard error to a file we use 2> to the end of the command, followed by the output error file you want to create. This sends the standard output stream to output1.txt, and the standard error stream to output2.err. The result is that, no output stream gets displayed in the console window. We can see the error messages by typing **output2.err**. This will open the file in your default text file viewer.
	Commands:

	dir file_name > output_file_name 2 > error_file_name.err
	Output Screenshot: 
Task 4	**Redirect All Output: Writes to Same File**
	Description: To include both the standard output stream and standard error streams to the same file, you just need to redirect all output to the same file using the following command. dir a.exe 1> output4.txt 2>&1 1. The standard output is directed to the output file identified by output number 1. 2. The standard error output identified by the number 2 is redirected to the output file identified by number 1. This will append the error output to the end of the standard output.
	Commands: dir a.exe 1> output4.txt 2>&1

	Output Screenshot: 
Task 5	**Input Redirection-Taking input from file**
	Description: To take input from a file, the less-than character < is used to redirect the input of a command. Command < file_name The commands that normally take their input from the standard input can have their input redirected from a file in this manner. For example, the sort command can take the input from the file as shown below. sort < input.txt input.txt file contains the numbers in an unsorted order
	Commands: sort < file_name
	Output Screenshot:



Task 6:	**Options in gcc compiler**

Description:

I) **-E** Stop after the preprocessing stage; do not run the compiler proper. The output is in the form of preprocessed source code, which is sent to the standard output. Input files that don't require preprocessing are ignored.

Example: $ gcc -E program_name.c

II) **-c** Compile or assemble the source files, but do not link. The linking stage simply is not done.The ultimate output is in the form of an object file for each source file. Example: $ gcc -c program_name.c

In the above case object file or executable wont be generated for the further execution.Only compilation will be done.

	III) -o Place output in file. This applies to whatever sort of output is being produced, whether it be an executable file, an object file, an assembler file or preprocessed C code. gcc program_name.c -o output
	Commands: Command for only preprocessing: gcc -E program_name.c Command for compiling without linking: gcc -c program_name.c Command for creating executable file: gcc program_name.c -o output
	Output Screenshot: **OUTPUT FOR ONLY PREPROCESSING COMMAND** **D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_1>gcc -E program1.c** **# 1 "program1.c"** **# 1 "<built-in>"** **# 1 "<command-line>"** **# 1 "program1.c"** **# 1 "c:\\mingw\\include\\stdio.h" 1 3** **# 38 "c:\\mingw\\include\\stdio.h" 3** **# 39 "c:\\mingw\\include\\stdio.h" 3** **# 56 "c:\\mingw\\include\\stdio.h" 3** **# 1 "c:\\mingw\\include_mingw.h" 1 3** **# 55 "c:\\mingw\\include_mingw.h" 3** **# 56 "c:\\mingw\\include_mingw.h" 3** **# 66 "c:\\mingw\\include_mingw.h" 3** **# 1 "c:\\mingw\\include\\msvcrtver.h" 1 3** **# 35 "c:\\mingw\\include\\msvcrtver.h" 3** **# 36 "c:\\mingw\\include\\msvcrtver.h" 3** **# 67 "c:\\mingw\\include_mingw.h" 2 3** **# 1 "c:\\mingw\\include\\w32api.h" 1 3** **# 35 "c:\\mingw\\include\\w32api.h" 3**

```
# 36 "c:\\mingw\\include\\w32api.h" 3
# 59 "c:\\mingw\\include\\w32api.h" 3
# 1 "c:\\mingw\\include\\sdkddkver.h" 1 3
# 35 "c:\\mingw\\include\\sdkddkver.h" 3

# 36 "c:\\mingw\\include\\sdkddkver.h" 3
# 60 "c:\\mingw\\include\\w32api.h" 2 3
# 74 "c:\\mingw\\include\\_mingw.h" 2 3
# 57 "c:\\mingw\\include\\stdio.h" 2 3
# 69 "c:\\mingw\\include\\stdio.h" 3
# 1 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stddef.h" 1 3 4
# 216 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stddef.h" 3 4

# 216 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stddef.h" 3 4
typedef unsigned int size_t;
# 328 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stddef.h" 3 4
typedef short unsigned int wchar_t;
# 357 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stddef.h" 3 4
typedef short unsigned int wint_t;
# 70 "c:\\mingw\\include\\stdio.h" 2 3
# 94 "c:\\mingw\\include\\stdio.h" 3
# 1 "c:\\mingw\\include\\sys/types.h" 1 3
# 34 "c:\\mingw\\include\\sys/types.h" 3

# 35 "c:\\mingw\\include\\sys/types.h" 3
# 62 "c:\\mingw\\include\\sys/types.h" 3
  typedef long __off32_t;


  typedef __off32_t _off_t;


  typedef _off_t off_t;
# 91 "c:\\mingw\\include\\sys/types.h" 3
  typedef long long __off64_t;


  typedef __off64_t off64_t;
# 115 "c:\\mingw\\include\\sys/types.h" 3
  typedef int _ssize_t;

  typedef _ssize_t ssize_t;
```

```
# 95 "c:\\mingw\\include\\stdio.h" 2 3


# 1 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stdarg.h" 1 3 4
# 40 "c:\\mingw\\lib\\gcc\\mingw32\\6.3.0\\include\\stdarg.h" 3 4
typedef __builtin_va_list __gnuc_va_list;
# 103 "c:\\mingw\\include\\stdio.h" 2 3
# 210 "c:\\mingw\\include\\stdio.h" 3
typedef struct _iobuf
{
  char *_ptr;
  int _cnt;
  char *_base;
  int _flag;
  int _file;
  int _charbuf;
  int _bufsiz;
  char *_tmpfname;
} FILE;
# 239 "c:\\mingw\\include\\stdio.h" 3
extern __attribute__((__dllimport__)) FILE _iob[];
# 252 "c:\\mingw\\include\\stdio.h" 3



 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * fopen (const char
*, const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * freopen (const
char *, const char *, FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fflush (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fclose (FILE *);



 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int remove (const char
*);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int rename (const char *,
const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * tmpfile (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) char * tmpnam (char *);
```

```
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) char *_tempnam (const
char *, const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _rmtmp (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _unlink (const char
*);
# 289 "c:\\mingw\\include\\stdio.h" 3
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) char * tempnam (const
char *, const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int rmtmp (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int unlink (const char *);



 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int setvbuf (FILE *, char
*, int, size_t);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) void setbuf (FILE *, char
*);
# 342 "c:\\mingw\\include\\stdio.h" 3
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,2,3))) __mingw_fprintf(FILE*, const
char*, ...);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,1,2))) __mingw_printf(const char*,
...);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,2,3))) __mingw_sprintf(char*, const
char*, ...);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,3,4))) __mingw_snprintf(char*,
size_t, const char*, ...);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,2,0))) __mingw_vfprintf(FILE*,
const char*, __builtin_va_list);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,1,0))) __mingw_vprintf(const char*,
__builtin_va_list);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,2,0))) __mingw_vsprintf(char*, const
char*, __builtin_va_list);
extern int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,3,0))) __mingw_vsnprintf(char*,
size_t, const char*, __builtin_va_list);
```

```
# 376 "c:\\mingw\\include\\stdio.h" 3
extern unsigned int _mingw_output_format_control( unsigned int, unsigned int );
# 453 "c:\\mingw\\include\\stdio.h" 3
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fprintf (FILE *, const
char *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int printf (const char *,
...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int sprintf (char *, const
char *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int vfprintf (FILE *,
const char *, __builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int vprintf (const char *,
__builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int vsprintf (char *, const
char *, __builtin_va_list);
# 478 "c:\\mingw\\include\\stdio.h" 3
 int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__ms_printf__,2,3))) __msvcrt_fprintf(FILE *, const
char *, ...);
 int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__ms_printf__,1,2))) __msvcrt_printf(const char *, ...);
 int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__ms_printf__,2,3))) __msvcrt_sprintf(char *, const
char *, ...);
 int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__ms_printf__,2,0))) __msvcrt_vfprintf(FILE *, const
char *, __builtin_va_list);
 int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__ms_printf__,1,0))) __msvcrt_vprintf(const char *,
__builtin_va_list);
 int __attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__ms_printf__,2,0))) __msvcrt_vsprintf(char *, const
char *, __builtin_va_list);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _snprintf (char *,
size_t, const char *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _vsnprintf (char *,
size_t, const char *, __builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _vscprintf (const char
*, __builtin_va_list);
# 501 "c:\\mingw\\include\\stdio.h" 3
```

```
__attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,3,4)))
int snprintf (char *, size_t, const char *, ...);

__attribute__((__cdecl__)) __attribute__((__nothrow__))
__attribute__((__format__(__mingw_printf__,3,0)))
int vsnprintf (char *, size_t, const char *, __builtin_va_list);

__attribute__((__cdecl__)) __attribute__((__nothrow__))
int vscanf (const char * __restrict__, __builtin_va_list);

__attribute__((__cdecl__)) __attribute__((__nothrow__))
int vfscanf (FILE * __restrict__, const char * __restrict__, __builtin_va_list);

__attribute__((__cdecl__)) __attribute__((__nothrow__))
int vsscanf (const char * __restrict__, const char * __restrict__, __builtin_va_list);
# 646 "c:\\mingw\\include\\stdio.h" 3
__attribute__((__cdecl__)) __attribute__((__nothrow__)) ssize_t
getdelim (char ** __restrict__, size_t * __restrict__, int, FILE * __restrict__);

__attribute__((__cdecl__)) __attribute__((__nothrow__)) ssize_t
getline (char ** __restrict__, size_t * __restrict__, FILE * __restrict__);
# 666 "c:\\mingw\\include\\stdio.h" 3
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fscanf (FILE *, const
char *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int scanf (const char *,
...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int sscanf (const char *,
const char *, ...);



 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fgetc (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) char * fgets (char *, int,
FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fputc (int, FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fputs (const char *,
FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) char * gets (char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int puts (const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int ungetc (int, FILE *);
# 687 "c:\\mingw\\include\\stdio.h" 3
```

```
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _filbuf (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _flsbuf (int, FILE *);



extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int getc (FILE *);
extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int getc (FILE * __F)
{
  return (--__F->_cnt >= 0)
    ? (int) (unsigned char) *__F->_ptr++
    : _filbuf (__F);
}

extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int putc (int, FILE *);
extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int putc (int __c, FILE * __F)
{
  return (--__F->_cnt >= 0)
    ? (int) (unsigned char) (*__F->_ptr++ = (char)__c)
    : _flsbuf (__c, __F);
}

extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int getchar (void);
extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int getchar (void)
{
  return (--(&_iob[0])->_cnt >= 0)
    ? (int) (unsigned char) *(&_iob[0])->_ptr++
    : _filbuf ((&_iob[0]));
}

extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int putchar(int);
extern inline __attribute__((__gnu_inline__)) __attribute__((__cdecl__))
__attribute__((__nothrow__)) int putchar(int __c)
{
  return (--(&_iob[1])->_cnt >= 0)
    ? (int) (unsigned char) (*(&_iob[1])->_ptr++ = (char)__c)
```

```
    : _flsbuf (__c, (&_iob[1]));}
# 734 "c:\\mingw\\include\\stdio.h" 3
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) size_t fread (void *,
size_t, size_t, FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) size_t fwrite (const void
*, size_t, size_t, FILE *);



 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fseek (FILE *, long,
int);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) long ftell (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) void rewind (FILE *);
# 787 "c:\\mingw\\include\\stdio.h" 3
typedef long long fpos_t;


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fgetpos (FILE *,
fpos_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fsetpos (FILE *, const
fpos_t *);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int feof (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int ferror (FILE *);
# 808 "c:\\mingw\\include\\stdio.h" 3
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) void clearerr (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) void perror (const char
*);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _popen (const
char *, const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _pclose (FILE *);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * popen (const char
*, const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int pclose (FILE *);
```

```
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _flushall (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _fgetchar (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _fputchar (int);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _fdopen (int,
const char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _fileno (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _fcloseall (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _fsopen (const
char *, const char *, int);

 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _getmaxstdio (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _setmaxstdio (int);
# 859 "c:\\mingw\\include\\stdio.h" 3
unsigned int __attribute__((__cdecl__)) __mingw_get_output_format (void);
unsigned int __attribute__((__cdecl__)) __mingw_set_output_format (unsigned
int);


int __attribute__((__cdecl__)) __mingw_get_printf_count_output (void);
int __attribute__((__cdecl__)) __mingw_set_printf_count_output (int);
# 885 "c:\\mingw\\include\\stdio.h" 3
extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
unsigned int __attribute__((__cdecl__)) _get_output_format (void)
{ return __mingw_get_output_format (); }

extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
unsigned int __attribute__((__cdecl__)) _set_output_format (unsigned int __style)
{ return __mingw_set_output_format (__style); }
# 910 "c:\\mingw\\include\\stdio.h" 3
extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
int __attribute__((__cdecl__)) _get_printf_count_output (void)
{ return 0 ? 1 : __mingw_get_printf_count_output (); }

extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
int __attribute__((__cdecl__)) _set_printf_count_output (int __mode)
{ return 0 ? 1 : __mingw_set_printf_count_output (__mode); }



 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fgetchar (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fputchar (int);
```

```
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * fdopen (int, const
char *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fileno (FILE *);
# 930 "c:\\mingw\\include\\stdio.h" 3
extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
FILE * __attribute__((__cdecl__)) __attribute__((__nothrow__)) fopen64 (const
char *, const char *);
extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
FILE * __attribute__((__cdecl__)) __attribute__((__nothrow__)) fopen64 (const
char * filename, const char * mode)
{ return fopen (filename, mode); }

int __attribute__((__cdecl__)) __attribute__((__nothrow__)) fseeko64 (FILE *,
__off64_t, int);

extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
__off64_t __attribute__((__cdecl__)) __attribute__((__nothrow__)) ftello64 (FILE
*);
extern inline __attribute__((__gnu_inline__)) __attribute__((__always_inline__))
__off64_t __attribute__((__cdecl__)) __attribute__((__nothrow__)) ftello64 (FILE *
stream)
{ fpos_t __pos; return (fgetpos(stream, &__pos)) ? -1LL : (__off64_t)(__pos); }
# 958 "c:\\mingw\\include\\stdio.h" 3
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fwprintf (FILE *,
const wchar_t *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int wprintf (const
wchar_t *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int vfwprintf (FILE *,
const wchar_t *, __builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int vwprintf (const
wchar_t *, __builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _snwprintf (wchar_t
*, size_t, const wchar_t *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _vscwprintf (const
wchar_t *, __builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _vsnwprintf (wchar_t
*, size_t, const wchar_t *, __builtin_va_list);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fwscanf (FILE *,
const wchar_t *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int wscanf (const wchar_t
*, ...);
```

```
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int swscanf (const
wchar_t *, const wchar_t *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t fgetwc (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t fputwc (wchar_t,
FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t ungetwc (wchar_t,
FILE *);

 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int swprintf (wchar_t *,
const wchar_t *, ...);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int vswprintf (wchar_t *,
const wchar_t *, __builtin_va_list);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wchar_t * fgetws
(wchar_t *, int, FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int fputws (const wchar_t
*, FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t getwc (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t getwchar (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t putwc (wint_t,
FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t putwchar
(wint_t);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wchar_t * _getws
(wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _putws (const
wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _wfdopen(int,
const wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _wfopen (const
wchar_t *, const wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _wfreopen (const
wchar_t *, const wchar_t *, FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _wfsopen (const
wchar_t *, const wchar_t *, int);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wchar_t * _wtmpnam
(wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wchar_t * _wtempnam
(const wchar_t *, const wchar_t *);
```

```
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _wrename (const
wchar_t *, const wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _wremove (const
wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) void _wperror (const
wchar_t *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * _wpopen (const
wchar_t *, const wchar_t *);

__attribute__((__cdecl__)) __attribute__((__nothrow__)) int snwprintf (wchar_t *,
size_t, const wchar_t *, ...);
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int vsnwprintf (wchar_t *,
size_t, const wchar_t *, __builtin_va_list);
# 1016 "c:\\mingw\\include\\stdio.h" 3
__attribute__((__cdecl__)) __attribute__((__nothrow__)) int vwscanf (const
wchar_t *__restrict__, __builtin_va_list);
__attribute__((__cdecl__)) __attribute__((__nothrow__))
int vfwscanf (FILE *__restrict__, const wchar_t *__restrict__, __builtin_va_list);
__attribute__((__cdecl__)) __attribute__((__nothrow__))
int vswscanf (const wchar_t *__restrict__, const wchar_t * __restrict__,
__builtin_va_list);

 __attribute__((__cdecl__)) __attribute__((__nothrow__)) FILE * wpopen (const
wchar_t *, const wchar_t *);

 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t _fgetwchar (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t _fputwchar
(wint_t);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _getw (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int _putw (int, FILE *);


 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t fgetwchar (void);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) wint_t fputwchar
(wint_t);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int getw (FILE *);
 __attribute__((__cdecl__)) __attribute__((__nothrow__)) int putw (int, FILE *);


# 2 "program1.c" 2

# 2 "program1.c"
```

```
int main()
{
 printf("Hello world");
 return 0;

}
```

```
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_1>gcc -c program1.c

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_1>gcc program1.c -o output5

D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_1>output5
Hello world
D:\PES\Semester 2\Computer_Science- C Programming\C_Lab\Week_1>
```