CSI 402 – Systems Programming

An Example for Make

Handout 1.3

File: `globals.h`

```
/* Global variables. */
int  p = 11; int  q = 32;
```

File: `externs.h`

```
/* Extern declarations for global variables. */
extern int  p;   extern int  q;
```

File: `main.c`

```
#include <stdio.h>
#include "globals.h"
int main(void) {
  /* Prototypes of two functions in a different file. */
  float  f1(int);   int    f2(char);
  /* Some local variables. */
  int i = 5; char c = 'x';
  /* Variables p and q used below are defined in "globals.h". */
  printf("Value of function 1 = %d\n", f1(i+p+q));
  printf("Value of function 2 = %d\n", f2(c));
  return  0;
} /* End of main. */
```

File: `funct.c`

```
/* This file contains the bodies of the functions used in main. */
#include "externs.h"

float  f1 (int t) {
   return ((float) t);
} /* End of f1. */

int  f2 (char w) {
   return p+q+(w-'a'+'A');  /* Uses global variables p and q. */
} /* End of f2. */
```

File: `makefile`

```
#The following rule tells make about possible suffixes
#(extensions) of file names.

.SUFFIXES: .c .o

#The following definition of CC ensures that
#gcc will be used to compile the C source files.

CC = gcc

#The following definition of CFLAGS ensures that
#the debugger can be used with the executable file (sample)
#created by running make.

CFLAGS = -g

#The following rule tells make how a ".o" file should
#be created from the corresponding ".c" file.
#Note that the "-c" option must be used here since we are
#compiling source files separately. (Note that the line
#following the ".c.o:" line begins with the "tab" character.)

.c.o:
        $(CC) $(CFLAGS) -c $<

#Dependency rule for the default target and how the
#default target is to be created. (Note that the line
#following the dependency rule begins with the "tab"
#character.)

sample: main.o  funct.o
        gcc main.o funct.o -o sample

#Dependency rules for other targets. (We don't need to
#specify how these targets are created since we have already
#given a general rule for creating a ".o" file from the
#corresponding ".c" file.)

main.o:  globals.h

funct.o: externs.h

#Target for removing unnecessary files.

clean:
        rm -f *.o core
```