Makefiles

To automate the compilation of the program with more than one source file we can use `make` command. Before this command will be able to automate the compilation process we need to create a `Makefile` with definitions how to exactly our program needs to be compiled

Example program divided into 3 files

```
`config.f90`
`sparse-algebra.f90`
`sparse-example.f90`
```

can be compiled using the following `Makefile`

```
FC=gfortran
FCFLAGS=-O2 -Wall
LIBS=
PROGRAM=sparse.x

SRC=config.o \
    sparse-algebra.o \
    sparse-example.o \

all: $(SRC)
  $(FC) $(FCFLAGS) $(SRC) $(LIBS) -o $(PROGRAM)


%.o: %.f90
  $(FC) $(FCFLAGS) -c $<

clean:
  rm -r *.o *.mod $(PROGRAM)
```

the first section

```
FC=gfortran
FCFLAGS=-O2 -Wall
LIBS=
PROGRAM=sparse.x
```

defines the commands that we will use later in the body of the Makefile, i.e. `FC` fortran

compiler, `FCFLAGS` flags to the compiler, `LIBS` external libraries to use, `PROGRAM` name of the executable.

```
SRC=config.o \
    sparse-algebra.o \
    sparse-example.o \
```

defines list of object files that we need to create. We have to define a rule which will desribe what needs to be done

```
all: $(SRC)
    $(FC) $(FCFLAGS) $(SRC) $(LIBS) -o $(PROGRAM)
```

and a rule how to compile source file to an object file

```
%.o: %.f90
    $(FC) $(FCFLAGS) -c $<
```

After the `Makefile` is prepared we can compile our program using `make` command.

```
$ make
gfortran -O2 -Wall -c config.f90
gfortran -O2 -Wall -c sparse-algebra.f90
gfortran -O2 -Wall -c sparse-example.f90
gfortran -O2 -Wall config.o sparse-algebra.o sparse-example.o   -o sparse.x
```