# KEY

- Individual Staff will help
- All code submitted is written by individual student
- You can get help from the teaching staff.

Read the specification. Completely.

Then read it again.

# Overview

Process chars, as they are read:

- Read chars from keyboard without echo
- If upper case A-Z write it to display
- If lower case a-z convert to upper case & write to display
    - subtract 20h
- If blank or period write it to display
- Anything else do not write to display
- If the char was a period then end the program

# Example

```
In: Hello, testing 123.
Out: HELLO TESTING .
```

# Step 1: design

- Pseudocode
  - Should not try your first ASM program in ASM
  - Sketch, outline the program in something you understand
  - C or Java are good pseudocode
- Readfile
  - Folder on website
  - Contains C & Java versions

# Step 2: code

- Retrieve testing and grading files from the key course locker
- Files in unpack.exe
- In DOSBox type: unpack
- Name your source code  key.asm
- Assemble
  - `ml /c /Zi /Fl key.asm`
  - Creates `key.obj`
- Link
  - `link /CO key.obj`
  - Creates `key.exe`

# Step 3: test

- testkey.exe
    - Runs your program (key.exe)
    - Redirects keyboard input to program
    - Logs your attempts

# Step 4: grade

- gradkey.exe
    - Self-grading
    - Produces file “key.ans”
- Grade
    - 60 pts — correctness
    - 20 pts — efficiency (number of instructions)
    - 20 pts — documentation

```
Times you used the grading program is 1
Grading program history
*** Grading Run. Date: 05/27/09
Test ran correctly.
++ Grade ++ 60 = Points for correct answers


We are now analyzing your source code.
20 points for 0 - 20
15 points for 21 - 22
10 points for 23 - 24
05 points for 25 - 26
00 points for 27+
Your actual counts are: Executables .. 20
++ Grade ++ 20 = Points for code written

Analyzing program documentation.
DOCPH-0027 program header
DOCHB-0.70 code block headers
DOCLC-0.95 line comments
++ Grade ++ 20 = Points for documentation

Calculating total grade
++ Grade ++ 100 = Total grade generated

Building key.ans for submission..
;----------------------------------
; Prog: KEY
```

```
Times you used the grading program is 1
Grading program history
*** Grading Run. Date: 05/27/09
Test ran correctly.
++ Grade ++ 60 = Points for correct answers


We are now analyzing your source code.
20 points for 0 - 20
15 points for 21 - 22
10 points for 23 - 24
05 points for 25 - 26
00 points for 27+
Your actual counts are: Executables .. 20
++ Grade ++ 20 = Points for code written

Analyzing program documentation.
DOCPH-0027 program header
DOCHB-0.70 code block headers
DOCLC-0.95 line comments
++ Grade ++ 20 = Points for documentation

Calculating total grade
++ Grade ++ 100 = Total grade generated

Building key.ans for submission..
;----------------------------------
; Prog: KEY
```

```
Times you used the grading program is 1
Grading program history
*** Grading Run. Date: 05/27/09
Test ran correctly.
++ Grade ++ 60 = Points for correct answers


We are now analyzing your source code.
20 points for 0 - 20
15 points for 21 - 22
10 points for 23 - 24
05 points for 25 - 26
00 points for 27+
Your actual counts are: Executables .. 20
++ Grade ++ 20 = Points for code written

Analyzing program documentation.
DOCPH-0027 program header
DOCHB-0.70 code block headers
DOCLC-0.95 line comments
++ Grade ++ 20 = Points for documentation

Calculating total grade
++ Grade ++ 100 = Total grade generated

Building key.ans for submission..
;----------------------------------
; Prog: KEY
```

```
Times you used the grading program is 1
Grading program history
*** Grading Run. Date: 05/27/09
Test ran correctly.
++ Grade ++ 60 = Points for correct answers


We are now analyzing your source code.
20 points for 0 - 20
15 points for 21 - 22
10 points for 23 - 24
05 points for 25 - 26
00 points for 27+
Your actual counts are: Executables .. 20
++ Grade ++ 20 = Points for code written

Analyzing program documentation.
DOCPH-0027 program header
DOCHB-0.70 code block headers
DOCLC-0.95 line comments
++ Grade ++ 20 = Points for documentation

Calculating total grade
++ Grade ++ 100 = Total grade generated

Building key.ans for submission..
;-----------------------------------
; Prog: KEY
```