Problem 1
The following program inputs a string terminated by return (newline) character. Then it capitalizes the alphabetic characters and prints the capitalized result. For example, if you input:

Hello123 world, hello

You will get as output:

HELLO123 WORLD, HELLO

Fill in the missing entries in the program. Note that \texttt{0dh} is the ascii code for return character.

\begin{verbatim}
code segment
  mov  _______,_________
  mov  ah,01h
loop1:
    int  21h
    mov  _______,al
    inc  bx
    cmp  _______,0dh
    jne  loop1
  mov  _______,_________
  mov  ah,02h
loop2:
  mov  _______,[bx]
  cmp  _______,'a'
  jb   _________
  cmp  _______,0dh
  ja   printit
  sub  ______,'a'
  add  dl,'A'
printit:
  inc  _______
  int  21h
  cmp  dl,_______
  jne  loop2
  int  _________
instring:

code ends
\end{verbatim}

Problem 2
The following A86 assembly language program inputs two strings and then searches the first string for the occurrence of a substring given by the second string. If the substring is found, then \texttt{y} is printed. If the substring is not found, then \texttt{n} is printed. You can assume the following about input strings:
- The first string is terminated by , (i.e. a comma).
- The second string is terminated by . (i.e. a period).
- The length of the first string is greater than or equal to the second string.
- The characters in the string are alphanumeric.

For example:

\texttt{ali,li.}

is an input which produces \texttt{y} answer. Fill in the missing entries in the program.
mov bp,______                        ; ( a )
mov ah,01h
mov dl,'n'

input:                     ; input strings
int 21h
mov _____,al                         ; ( b )
cmp al,'.'
jne notcomma

jmp notcomma:
inc bp
___   input                            ; ( d )

endinp:                     ; end of input
mov bp,______                        ; ( e )
dec bp

loop1:
inc bp
cmp _____,','                         ; ( f )
je nomatch
xor di,di         ; set di to 0

loop2:
cmp __________,','                      ; ( g )
jne loop1
inc di
jmp loop2

yesmatch:
mov dl,'y'

nomatch:
mov ____,___                         ; ( j )

int 21h
int 20h       ; exit to dos

arry: db ?

code ends