include Irvine32.inc

.data

letter byte ?

string byte "Enter a leter :",0

string1 byte "Enter + or - ",0

array byte 5 dup (0)

len dword lengthof string

cha byte ?

vol byte "Vowel :",0

svol dword lengthof vol

check byte ?

up byte ?

curr byte "current vowels array : ",0

rev byte "Reverse :",0

.code

main proc

call start

call sec

call che

call lest

call last

invoke ExitProcess, 0

main endp

start proc

mov edx , offset string

call writestring

ret

start endp

sec proc

call readchar

mov cha , al

call readchar

call writechar

call crlf

mov edx , offset string1

call writestring

ret

sec endp

three proc

call crlf

.if al >= 65

.while (al <= 90)

.if (al == 'A') || (al == 'E') || (al == 'I') || (al == 'O') || (al == 'U')

mov edx , offset vol

mov ecx , svol

call writestring

.endif

call writechar

add al , 32

call writechar

sub al , 32

inc al

call crlf

.endw

.endif

ret

three endp

three1 proc

.if al >= 65

.while (al >= 65)

.if (al == 'A') || (al == 'E') || (al == 'I') || (al == 'O') || (al == 'U')

mov edx , offset vol

mov ecx , svol

call writestring

.endif

call writechar

add al , 32

call writechar

sub al , 32

dec al

call crlf

.endw

.endif

ret

three1 endp

che proc

.if (al >= 65 ) && (al <= 90)

mov up , al

call readchar

.if (al == '+')

mov al , up

call three

.elseif (al == '-')

mov al , up

call three1

.endif

.elseif (al >= 97 ) && (al <= 122)

mov up , al

call readchar

.if (al == '+')

mov al , up

call fo

.elseif (al == '-')

mov al , up

call fo1

.endif

.endif

ret

che endp

fo proc

call crlf

.if al >= 97

.while (al <= 122)

.if (al == 'a') || (al == 'e') || (al == 'i') || (al == 'o') || (al == 'u')

mov edx , offset vol

mov ecx , svol

call writestring

.endif

call writechar

sub al , 32

call writechar

add al , 32

inc al

call crlf

.endw

.endif

ret

fo endp

fo1 proc

.if al >= 97

.while (al >= 97)

.if (al == 'a') || (al == 'e') || (al == 'i') || (al == 'o') || (al == 'u')

mov edx , offset vol

mov ecx , svol

call writestring

.endif

call writechar

sub al , 32

call writechar

add al , 32

dec al

call crlf

.endw

.endif

ret

fo1 endp

lest proc

call crlf

mov edx , offset curr

mov eax , lengthof curr

call writestring

call crlf

mov edx , offset rev

mov eax , lengthof rev

call writestring

call crlf

ret

lest endp

last proc

call crlf

call waitmsg

ret

last endp

end main