Código genera	do por	gcc con -O2 pa	ra Intel x86																
		instrucción	operandos	explicación	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	1	movapd	x(%rax), %xmm0	; xmm0<-M[x+rax]															
	2	movapd	x+16(%rax), %xmm2	; xmm2<-M[x+16+rax]															
Etapas	3	addq	\$32, %rax	; rax=rax+32															
cauce	4	mulpd	%xmm1, %xmm0	; xmm0=xmm0*xmm1															
Se procesan	5	mulpd	%xmm1, %xmm2	; xmm2=xmm2*xmm1															
1 vez	6	movapd	y-32(%rax), %xmm3	; xmm3<-M[y-32+rax]															
las instruc.	7	movapd	y-16(%rax), %xmm4	; xmm4<-M[y-16+rax]															
1 a 12	8	addpd	%xmm3, %xmm0	; xmm0=xmm0+xmm3															
	9	addpd	%xmm4, %xmm2	; xmm2=xmm2+xmm4															
	10	movaps	%xmm0, y-32(%rax)	; M[y-32+rax]<-xmm0															
	11	movaps	%xmm2, y-16(%rax)	; M[y-16+rax]<-xmm2															
	12	cmpq	%rdx, %rax	; eax-edx															
bits válido	1	movapd	x(%rax), %xmm0	; xmm0<-M[x+rax]	1														
ventana(s)	2	movapd	x+16(%rax), %xmm2	; xmm2<-M[x+16+rax]	1														
instrucciones	3	addq	\$32, %rax	; rax=rax+32	1														
0	4	mulpd	%xmm1, %xmm0	; xmm0=xmm0*xmm1	10														
estación o	5	mulpd	%xmm1, %xmm2	; xmm2=xmm2*xmm1	10														
estaciones	6	movapd	y-32(%rax), %xmm3	; xmm3<-M[y-32+rax]	0														
de reserva	7	movapd	y-16(%rax), %xmm4	; xmm4<-M[y-16+rax]	0														
(al finalizar	8	addpd	%xmm3, %xmm0	; xmm0=xmm0+xmm3	00														
el ciclo)	9	addpd	%xmm4, %xmm2	; xmm2=xmm2+xmm4	00														
	10	movaps	%xmm0, y-32(%rax)	; M[y-32+rax]<-xmm0	00														
	11	movaps	%xmm2, y-16(%rax)	; M[y-16+rax]<-xmm2	00														
	12	cmpq	%rdx, %rax	; eax-edx	10														

movapd	x(%rax), %xmm0	1 bit de válido porque usa un registro con datos de entrada (rax), en xmm0 se carga el dato que se trae de memoria.
movaps	%xmm0, y-32(%rax)	2 bits de válido porque usa dos registros con datos de entrada (xmm0, y rax para calcular la dirección en la que se escribe). El primer bit se refiere a xmm0, el segundo a rax (siguiendo el orden en el que aparecen en la instrucción)
M[y-16+rax]		nota acceso a memoria a la dirección que se obtiene de la siguiente operación y-16+rax
mulpd	%xmm1, %xmm0	bits de válido inicialmente 10 porque el operando en %xmm1 está disponible pero el operando en %xmm0 no lo está (lo carga la instr. (1))
Posible pregu	nta en el control : ¿Qué ins	trucciones se están ejecutando en el ciclo 6?
Para contesta	r a las preguntas hay que re	ellenar la tabla primero.