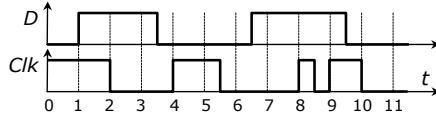


Cognome _____ Matricola: _____ Com. Dig.
e nome: _____ Inf. Mus.

1. [2] Che numero decimale rappresenta il codice esadecimale: **0x43B48000** nel formato standard IEEE-754 a singola precisione?
2. [5] Si disegni la struttura circuitale interna di un **flip-flop** di tipo D e si disegni l'andamento dell'uscita Q in conseguenza degli ingressi rappresentati in figura. Si trascurino i ritardi di porta e sia $Q(0)=0$.



3. [5] Si progetti un circuito caratterizzato da un ingresso a 4 bit A: ($a_3 a_2 a_1 a_0$) e da un'uscita a 2 bit Y: ($y_1 y_0$) che deve valere:
 '11' se: $a_3=1$; '10' se: $a_3=0$ e $a_2=1$;
 '01' se: $a_3=0$, $a_2=0$ e $a_1=1$; '00' se: $a_3=0$, $a_2=0$ e $a_1=0$;
 a) Determinare la tabella di verità di Y; b) esprimerla nella forma canonica più adatta; c) semplificarla mediante mappe di Karnaugh; d) semplificarla ulteriormente, se possibile, mediante semplificazioni algebriche; e) disegnarne lo schema circuitale.
4. [8] Si sintetizzi una macchina a stati finiti di Moore caratterizzata da una linea di ingresso che viene valutata ogni decimo di secondo e da una linea di uscita la quale, ad ogni fronte di salita all'ingresso, va a '1' e ci rimane per 3 colpi di clock, e poi torna a '0'. Si supponga per semplicità che tra due fronti di salita consecutivi all'ingresso passi sempre almeno 1 secondo.
 Si determinino: STG, STT, STT codificata e struttura circuitale del sistema completo, non trascurando la gestione del segnale di clock ed avendo cura di semplificare il più possibile le funzioni prima di tradurle in circuito.

5. [6] Si traduca in linguaggio Assembly MIPS nativo, evitando cioè di utilizzare pseudoistruzioni, la seguente procedura in linguaggio C. La procedura si aspetta l'argomento **n** nel registro **\$a0** e restituisce il risultato in **\$v0**.

```
int MegaFunz ( int n )
{
    if (n<3)    return ( n );
    else       return ( (n-2)*MegaFunz (n-1) );
}
```

6. [4] Si scriva un programma Assembly completo, in ambiente SPIM, che calcola il valore di "MegaFunz" di un numero fornito da tastiera chiamando la funzione sviluppata all'esercizio precedente, e quindi termina. Il programma deve presentarsi a terminale come nell'esempio a lato:

```
Inserisci l'argomento
> 23
Megafunz ( 23 ) = 46
```

7. [4] Rappresentare il contenuto byte per byte, in formato esadecimale, e gli indirizzi corrispondenti della zona di memoria che viene modificata a seguito dell'esecuzione del segmento dati riportato a lato. Si ricorda che il codice ASCII della lettera "A" è, in decimale, 65.

```
.data 0x1234
.asciiz "ABC"
.half -45,45
.byte +9,-9
.word 0x1234
```

System calls

	codice (\$v0)	argomenti	risultato
print int	1	\$a0	
print float	2	\$f12	
print double	3	\$f12	
print string	4	\$a0	
read int	5		\$v0
read float	6		\$f0
read double	7		\$f0
read string	8	\$a0,\$a1	
sbrk	9	\$a0	\$v0
exit	10		

Registri MIPS

0	zero	24-25	t8 - t9
1	at	26-27	k0 - k1
2-3	v0 - v1	28	Gp
4-7	a0 - a3	29	Sp
8-15	t0 - t7	30	s8
16-23	s0 - s7	31	Ra

MIPS Instruction Set:

