Assignment 2
Due date: 11th February, 2015
Maximum Marks: 100

Introduction:

Design a digital calendar module. The calendar gives the current date as an output. Use counters to keep track of the date in a digital calendar.

Concept:

A counter is a device which stores the number of times a particular event or process has occurred, often in relationship to a clock signal.
On exceeding its maximum value, the counter reverts back to its original value.

Three counters will be used - a Date counter, a Day counter, and a Month Counter. The Date counter ideally receives a clock signal as input.
When its value is 30(or 31 or 28, depending on the month counter), on receiving the next clock pulse, it resets itself to 1 and sends a clock signal to the Month counter.
Day counter also receives a clock signal as input.
When its value is 7, on receiving the next clock pulse, it resets itself to 1.
For Month counter increment the value on every clock signal received from Date counter and resets itself to 1 on reaching the value 12.
Note: Assume 28 days in February.

Implementation:

For this problem,
- use the following clock design

```
      reg clock
      always begin
          #5  clock =~clock; // Toggle clock every 5 ticks
      end
```

- synthesize a DigitalCalendar module with following input/output parameters,

Input: day[2:0], date[4:0], and month[3:0] (assume that input will be a valid date - e.g. day = 4(corresponds to thursday), date = 1, month = 1), and a clock. Changes the date/day on every positive edge of the clock.
Output: the current Day[2:0], Date[4:0], and Month[3:0]

Testbench Module instructions:

Test for validity of output after 10, 100, and 1000 positive edges of the input clock.