#import<Foundation/Foundation.h>  
#include<stdlib.h>  
#include<time.h>  
  
int main(int argc, const char\* argv[])  
{  
 NSAutoreleasePool \*pool = [[NSAutoreleasePool alloc]init];  
 NSTimeInterval interval = 24\*60\*60, random = 0;  
 NSDate \*today = [NSDate date];  
 NSMutableArray \*dates = [[NSMutableArray alloc]init];//For storing randomly generated dates  
  
 //today's date  
 //DateFormatter to remove time part of date.  
 NSDateFormatter \*dateFormat = [[NSDateFormatter alloc] init];  
 [dateFormat setDateFormat:@"dd-MM-yyyy"];  
 NSString \*dateString = [dateFormat stringFromDate:today];  
 NSLog(@"Today's date: %@", dateString);  
  
 //Day after tomorrow's date  
 NSDate \*dayAfterTomorrow = [[NSDate alloc] initWithTimeIntervalSinceNow:24\*60\*60\*2];  
 NSString \*dayAfterTomorrowString = [dateFormat stringFromDate:dayAfterTomorrow];  
 NSLog(@"Day after tomorrow's date: %@", dayAfterTomorrowString);  
  
 //Last thursday's date  
 NSCalendar \*gregorian = [[[NSCalendar alloc] initWithCalendarIdentifier:NSGregorianCalendar] autorelease];  
 NSDateComponents \*component = [gregorian components:NSWeekdayCalendarUnit fromDate:today];  
 int weekday = [component weekday];  
 if(weekday>5) //to check if we have passed thurday in the current week  
 {  
 interval = 24\*60\*60\*(weekday-5);  
 }  
 else  
 {  
 interval = 24\*60\*60\*(weekday+2); //+2 for saturday and friday of last week.  
 }  
 NSDate \*lastThursday = [[NSDate date] addTimeInterval:-interval];  
 NSString \*lastThursdayString = [dateFormat stringFromDate:lastThursday];  
 NSLog(@"Last Thursday's date: %@",lastThursdayString);  
  
 //finding earliest date among a given set of dates  
 int i=0;  
 interval = 24\*60\*600;   
 NSDate \*randomDate = nil;  
 NSLog(@"Randomly generated dates: \n");  
 for(i=0;i<5;i++)  
 {  
 random = interval\* (rand()%5);  
 randomDate = [[NSDate alloc] initWithTimeIntervalSinceNow:random];  
 [dates addObject:randomDate];  
 NSString \*randomDateString = [dateFormat stringFromDate:randomDate];  
  
  
 NSLog(@"%@", randomDateString);  
 }  
 NSDate \*earliestDate = nil;  
 for(id entry in dates)  
 {  
  
 if(earliestDate == nil)  
 {  
 earliestDate = entry;  
 }  
 else if([earliestDate compare:entry] == NSOrderedDescending)  
 {  
 earliestDate = entry;  
 }  
 }  
 NSString \*earliestDateString = [dateFormat stringFromDate:earliestDate];  
 NSLog(@"The earliest date among the above given dates: %@", earliestDateString);  
  
  
 //finding tenth day of the month given the first day  
   
 NSDictionary \*days = [NSDictionary dictionaryWithObjectsAndKeys:  
 @"sunday", [NSNumber numberWithInteger:1],  
 @"monday", [NSNumber numberWithInteger:2],  
 @"tuesday", [NSNumber numberWithInteger:3],  
 @"wednesday", [NSNumber numberWithInteger:4],  
 @"thurday", [NSNumber numberWithInteger:5],  
 @"friday", [NSNumber numberWithInteger:6],  
 @"saturday", [NSNumber numberWithInteger:7], nil];  
  
  
 id firstDayOfMonth = [NSNumber numberWithInteger:1];//Assuming 1st day is sunday according to Dictionary assigned  
 int tenthDay = ([firstDayOfMonth intValue]+2)%7; //Calculating the value of 10th day to look in dictionary  
 id tenthDayOfMonth = [NSNumber numberWithInteger:tenthDay];  
 NSLog(@"First day of the month is %@,tenth day is %@",[days objectForKey:firstDayOfMonth],[days objectForKey:tenthDayOfMonth]);  
  
 [pool release];  
 return 0;  
}