# Northern Michigan University (Marquette Co, MI)

CS344-01-24F: iOS / iPhone Programming (Andrew A. Poe) Name: _____
Endterm Examination (Exam 2) Page 1/2 Friday 5 December 2025 3:00 P.M. EST

Time: 50 minutes.

1. Write Swift code that uses an sqlite database (already open) with a table called NMURegistration containing fields named StudentName and ClassName(both are strings). Your Swift code should print (to the screen) the names of all of the students enrolled in either CS344 or CS202.

```
sqlite3_prepare_v2(db,
"select StudentName from NMURegistration where ClassName='CS344'
or ClassName='CS202'",-1,&stmt,nil)
while sqlite3_step(stmt)==SQLITE_ROW {
  let n:String = String (cString:sqlite3_column_text(stmt,0));
  print ("\(n)")
}
sqlite3_reset (stmt)
sqlite3_finalize (stmt)
```

2. In Main.storyboard, I have a tab bar controller controlling two view controllers. I have a button in the first view controller and a label in the second view controller.

Write the Swift @IBAction method for the button in the first view controller. When the button is clicked, the label in the second view controller should contain the total number of times the button was clicked since the app was launched.

```
var clickcount:Int = 0
@IBAction func Increment () {

   clickcount += 1
   svc!.lab!.text = "\(clickcount)"
 }
```

3. I have a TableViewController that never changes. (You don't have to worry about arrays or databases since the Table never changes. Also don't worry about clicking on an entry.) It has ten sections labeled "1" through "10". The first section has 26 rows, one for each letter of the alphabet "A"..."Z". The second section has rows for the alphabet twice. "A"..."Z" followed by another "A"..."Z". The third section has rows for the alphabet three times, etc.

Write the code for the following four methods that will cause the table to display correctly. Think about these methods carefully.

```
    func numberOfSections(in tableView: UITableView) -> Int

{return 10}
```

```
    func tableView(_ tableView: UITableView, numberOfRowsInSection section:
Int) -> Int

{return 26*(section+1)}

    func tableView(_ tableView: UITableView, cellForRowAt indexPath:
IndexPath) -> UITableViewCell

    {
        let CellIdentifier = "alphabet"
        let cell = tableView.dequeueReusableCell(withIdentifier:
CellIdentifier, for: indexPath)
        let
alph:[String]=["A","B","C","D","E","F","G","H","I","J","K","L","M","N","O","P
","Q","R","S","T","U","V","W","X","Y","Z"]
        let row = indexPath.row
        cell.textLabel!.text = alph[row%26]

        return cell
    }


    func tableView(_ tableView: UITableView, titleForHeaderInSection section:
Int) -> String?

{return ("\(section+1)")
```

4. Write the viewDidLoad for a simple app (no navigation or tab controller or anything like that; just a regular view on a view controller). The viewDidLoad method writes to a label that already exists and is already connected to the storyboard. Your method should write to the label the latitude and longitude in degrees and minutes. For example, the coordinates of Marquette are "46n33 87w24". The degrees come before the letter and the minutes come after it.

Write the method below as well.

```
func locationManager (_ manager:CLLocationManager,didUpdateLocations
locations:[CLLocation]) { ... }

        let loc = locations[0]
        let coord = loc.coordinate
        let longitude  = coord.longitude
        let latitude = coord.latitude
        var NS = "N"
        if latitude < 0 {NS = "S" ; latitude = -latitude}
        var EW = "E"
        if longitude < 0 {EW = "W" ; longitude = -longitude}
        let latd = floor (latitude)
        let longd = floor (longitude)
        let latm = round (60*(latitude-latd))
        let longm = round (60*(longitude-longd))
```

```
        lab!.text = "\(latd)\(NS)\(latm) \(longd)\(EW)\(longm)"
}

    override func viewDidLoad() {
        super.viewDidLoad()
        locationManager.delegate = self
        locationManager.desiredAccuracy = kCLLocationAccuracyBest
        locationManager.requestAlwaysAuthorization()
        locationManager.startUpdatingLocation()
    }
```