

```
//N2N-getLocation.swift
/**
 * Gets the location from the device's GPS sensor.
 * @param context The context of the application
 * @return String GPS location
 */
func getLocation(/*unknown token*/) -> String {
    //Insert Native-2-Native result
    var latitude:String = "" /*unknownToken*/
    var longitude:String = "" /*unknownToken*/
    return latitude + ", " + longitude
}

/* NATIVE-2-NATIVE RESULT
override func viewDidLoad() {
    super.viewDidLoad()
    // Do any additional setup after loading the view,
    typically from a nib.
    locationManager = CLLocationManager()
    locationManager.delegate = self
    locationManager.desiredAccuracy = kCLLocationAccuracyBest
    locationManager.requestAlwaysAuthorization()
    locationManager.startUpdatingLocation()
}
*/
```

```
//N2N-putHashMap.swift
/**
 * Associates the specified value with the specified key in this map. If the
 * map previously contained a mapping for the key, the old value is
 * replaced.
 */
func putHashMap(/*unknown token*/, B: String, C: String) -> /*unknown token*/ {
    //Insert Native-2-Native result
    var putHashMapTemp:/*unknown token*/ = A
    putHashMapTemp./*unknown token*/(B,C)
    return putHashMapTemp
}

/* NATIVE-2-NATIVE RESULT
extension String {
    var md5: String! {
        let str = self.cStringUsingEncoding(NSUTF8StringEncoding)
        let strLen = CC_LONG(self.lengthOfBytesUsingEncoding(NSUTF8StringEncoding))
        let digestLen = Int(CC_MD5_DIGEST_LENGTH)
        let result = UnsafeMutablePointer<CUnsignedChar>.alloc(digestLen)
        CC_MD5(str!, strLen, result)
        var hash = NSMutableString()
        for i in 0..

```