INFO 7500 – HW2
WENQING LIANG
001873144

Part 2

A. Yes, as long as the seed used in SHA1PRNG is strong enough to create lots of randomness in the output.

B. If the programmer is specifying a weak seed for PRNG, it may be easy for the attacker to predict since the randomness may be traceable. The output (e.g.: key pairs, etc.) can be extrapolated and safety of the system can be jeopardized.

C. The getInstanceStrong() method uses the strong algorithm specified in java.security file, which is [securerandom.strongAlgorithms=NativePRNGBlocking:SUN]. In the new NativePRNGBlocking implementation, getting output is not thread-safe, so using it for web application to generate key pairs may create problems.

Part 4 (see code file)

public key:

-----BEGIN PUBLIC KEY-----
MFYwEAYHKoZIzj0CAQYFK4EEAAoDQgAEQdFBrJohzYajfUBh4S1i+0y6oxXZMBm2
5iwzfsieVyKNUhBJylntm4KrhsXS0hod9KYQjiDIcBKYsA8y5v1blw==
-----END PUBLIC KEY-----

Part 5 (see code file)

digital signature hex:

3045022026604AEB264351A8F913A6698F9A0608D871AABDACE89B09FB43C009E4677A
CB022100B847E173A69FB746EAF59C1C01C7CC19147E9A1AFDD6BE1B4BA9392C97C817
2B