學號：r08944035 姓名：呂翊愷

d = 944035

1. (10338857399563508035974916425421659830878883530402360147780309523428
6494993683,
37057141145242123013015316630864329550140216928701153669873286428255828810018)
2. (21505829891763648114329055987619236494102133314575206970830385799158076338148,
98003708678762621233683240503080860129026887322874138805529884920309963580118)
3. (50378940446887064391710087938733604239839275616425037950555357248360521121569,
38772757134731078236553882891412268928481957188881790460577003933187089550225)
4. 944035(binary): 11100110011110100011
   將944035換成2進位後，從前面第2個bit開始，遇到1就double&add，遇到0就double，直到最後一個bit。
   add: 11, double: 19
5. 944035(binary): 11100110011110100011
   將944035換成2進位後，從前面第2個bit開始，若後面有連續n個bit (n > 1)為1，先add一次，再double n次，再subtract。若後面只有1個bit為1，則double 1次且add 1次。若後面的bit為0，則double 1次即可。
   add: 5, double: 19, subtract: 4
6.

```
# problem 6
def ExtendedEuclidean(n, m):
    if (m == 0):
        return 1, 0
    else:
        x, y = ExtendedEuclidean(m, n % m)
        x, y = y, (x - (n // m) * y)
        return x, y

def ModularInverse(k, n):
    return ExtendedEuclidean(k, n)[0]

dA = 944035
QA = dA * G

z = 0x38316DC32F31B3BC25DC18A61E682E86837877689209A3EC1562CE59E47CE13B
k = 228
P = k * G
n = G.order()

kInv = ModularInverse(k, n)
r = Mod(P[0], n)
s = Mod(kInv * (z + r * dA), n)
print("r = ", r)
print("s = ", s)
```

Result:

```
('r = ', 57165057262739177729018718925418755136619440075372687527386120109554898178633)
('s = ', 73178881436934743067295953878641018213422989279699072266236471750698068829831)
```

7.

```
# problem 7
if r < 1 and r > n-1:
    print "invalid r"
if s < 1 and s > n-1:
    print "invalid s"

sInv = ModularInverse(int(s), n)
w = Mod(sInv, n)
u1 = Mod(z * w, n)
u2 = Mod(r * w, n)
L = int(u1) * G + int(u2) * QA

if Mod(r, n) == int(L[0]):
    print "Signature valid"
```

Result:

```
Signature valid
```