Candidate Number: 2031B
Paper 3
Question 3

a.

| | | |
|---|---|---|
| 2 | 3 | 1 |
| 4 | 3 | 2 |
| 5 | 4 | 2 |

```
def f(a,b,x,y):
    return (a[1]-b[1])*x+(b[0]-a[0])+a[0]*b[1]-b[0]*a[1]

def lookup_a(x, y, V, T):
    for (i, j, k) in T:
        alpha = f(V[k], V[j], x, y)/f(V[k],V[j], V[i][0], V[i][1])
        beta = f(V[i], V[k], x, y)/f(V[i],V[k], V[j][0], V[j][1])
        gamma = 1 - alpha - beta
        if (0 <= alpha <= 1 and 0 <= beta <= 1 and 0 <= gamma <= 1):
            return alpha * V[i][2] + beta * V[j][2] + gamma *
V[k][2]

    return -1
```

b.

```
def f(a,b,x,y):
    return (a[1]-b[1])*x+(b[0]-a[0])+a[0]*b[1]-b[0]*a[1]


def lookup_a(x, y, V, T):
    min_depth = -Infinity
    a = -1
    for (i, j, k) in T:
        alpha = f(V[k], V[j], x, y)/f(V[k],V[j], V[i][0], V[i][1])
        beta = f(V[i], V[k], x, y)/f(V[i],V[k], V[j][0], V[j][1])
        gamma = 1 - alpha - beta
        if (0 <= alpha <= 1 and 0 <= beta <= 1 and 0 <= gamma <= 1):
            z = alpha * V[i][2] + beta * V[j][2] + gamma * V[k][2]
            if (z < min_depth):
                a = alpha * V[i][3] + beta * V[j][3] + gamma *
V[k][3]
                min_depth = z

    return a
```

c.