1.

(a) True
(b) True
(c) True
(d) False
(e) True
(f) False
(g) True

2.

(a)
```
for i = 1 to A.length - 1
    min = i
    for j = i+1 to A.length
        if A[j] < A[min]
            min = j
    A[i], A[min] = A[min], A[i]
```

(b) Before the start of each iteration of the outer loop, the subarray from A[1] to A[i-1] contains the smallest elements of the original array and is sorted. The rest of the array contains the remaining elements in no particular order.

(c) The best-case and worst-case running times of selection sort are both $\Theta(n^2)$ since every element is compared with every other element.

3.

(a)
```
def MOVE(n, start, end):
    if n==1:
        PRINT(start, end)
    else:
        mid = 6-start-end
        MOVE(n-1, start, mid)
        PRINT(start, end)
        MOVE(n-1, mid, end)
```

(b) $f(n) = 2f(n-1) + 1$

Since $f(1) = 1 = 2^1 - 1$

$f(2) = 2f(1) + 1 = 2^2 - 2 + 1 = 2^2 - 1$

$f(3) = 2f(2) + 1 = 2^3 - 2 + 1 = 2^3 - 1$

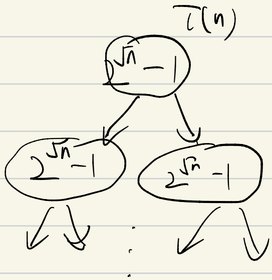
$\cdots\cdots$

$f(n) = 2^n - 1$

4.

(a) $T(n) = 2T(n-a) + T(a)$

(b) $a = \sqrt{n}$ $\qquad (\qquad)(2^{n}-1)$



$L=0$ $\qquad 2^{\sqrt{n}}-1$

$L=1$ $\qquad 2(2^{\sqrt{n}}-1)$

$\vdots$

$L=\sqrt{n}-1$ $\qquad 2^{\sqrt{n}-1}\cdot(2^{\sqrt{n}}-1)$

$$
\begin{aligned}
T(n) &= (2^{\sqrt{n}}-1)\cdot(2+4+8+\cdots+2^{\sqrt{n}-1}) \\
&= (2^{\sqrt{n}}-1)\,(2^{\sqrt{n}}-2) \\
&\le 2^{\sqrt{n}}\cdot 2^{\sqrt{n}} = 2^{2\sqrt{n}}
\end{aligned}
$$

Therefore, $c=2$ is sufficient for $T(n) \le 2^{c\sqrt{n}}$