Marking for Theory Assignment 1
maximum 18 marks

[3] Question 1

Lots of different ways to write this. Here's one that captures what needs to be done, I think, but we just evaluated what you gave us...it didn't have to look like this.

```
function array_sum(begin, end, x)
  if begin > end            // have run out of array
    return 0
  else if array[begin] < x  // array element at index begin is added
    return array[begin] + array_sum(begin + 1, end, x)
  else                      // array element is not added
    return array_sum(begin + 1, end, x)
```

[3] Question 2
a. Since there are n! possible tours and for every iteration it takes O(n) to add up the paths, we have T(n) = n! * O(n), or O(n!) [2 marks]
b. 10! : 1 sec yields 20! : ? sec, or 20! / 10! (about 20000 years) [1 mark]

[4] Question 3

Sorry, no fancy math symbols here. Pretend 'sum' is a big sigma symbol, and so on. Here's one way (of many) to do it...

```
sum(3->n) (k^2 - 2k)
= sum(1->n) (k^2 - 2k) - sum(1->2) (k^2 - 2k)
= (n(n+1)(2n+1))/6 - 2(n(n+1))/2 - (1-2+4-4)
=(2n^3 + 3n^2 + n)/6 - (n^2 + n) + 1
=(n^3)/3 + (n^2)/2 + n/6 - n^2 - n + 1
= (n^3)/3 - (n^2)/2 - 5n/6 + 1

T(n) = (n^3)/3 - (n^2)/2 - 5n/6 + 1
guess that lower bound is cn^3 for all n > some n0

Find some c and n0 such that
(n^3)/3 - (n^2)/2 - 5n/6 + 1 > cn^3 for all n > n0

(n^3)/3 - (n^2)/2 - 5n/6 + 1 >= cn^3
6(n^3)/3 - 6(n^2)/2 - 6*5n/6 + 6*1 >= 6cn^3
2(n^3) - 3(n^2) - 5n + 6 >= 6cn^3
let c = 1/6
2(n^3) - 3(n^2) - 5n + 6 >= n^3
let n0 = 5
2*125 - 3*25 - 5*5 + 6 >= 125
156 > 125
try another n0 -- let n0 = 6
2*216 - 3*36 - 5*6 + 6 >= 216
300 >= 216
the difference is increasing as n gets bigger, so we should be ok
```

(there are other ways to assign c and n0 that will work, of course)

[8] Question 4. [2 marks each:]

a O(1) & Omega(1).
b O(n), c = 5, n0 = 1; Omega(n), c = 3
c O(n^3), c = 6, n0 = 1; Omega(n^3), c = 3
d O(n lg n), c = 4; Omega(n lg n), c=1, n0 = 1