Maximum contiguous subsequence sum problem — Sol. 1

Problem: Find $\max\left\{ \sum_{i=\ell}^{h-1} a[i] \;\middle|\; 0 \le \ell \le h \le n \right\}$

where $a[0\,..\,n-1]$ is an array of integers (possibly negative) and $n \ge 0$.

Solution 1:

```
summax := 0;
for ℓ from 0 to n − 1 :
  for h from ℓ + 1 to n :
    {
      sum := 0;
      for i from l to h − 1 :
        sum := sum + a[i];
      if summax < sum then summax := sum;
    }
```

Maximum contiguous subsequence sum problem — complexity of Sol. 1

$$
\begin{aligned}
c_1 + \sum_{\ell=0}^{n-1} \left(c_2 + \sum_{h=\ell+1}^{n} \left(c_3 + \sum_{i=\ell}^{h-1} c_4 \right) \right) \;&=\; c_1 + \sum_{\ell=0}^{n-1} \left(c_2 + \sum_{h=\ell+1}^{n} (c_3 + c_4 \cdot (h - \ell)) \right) \\
&= c_1 + \sum_{\ell=0}^{n-1} \left(c_2 + \frac{((c_3 + c_4) + (c_3 + c_4 \cdot (n - \ell))) \cdot (n - \ell)}{2} \right) \\
&= c_1 + c_2 \cdot n + \left(\sum_{\ell=0}^{n-1} \frac{(2 \cdot c_3 + c_4) \cdot (n - \ell)}{2} \right) + \left(\sum_{\ell=0}^{n-1} \frac{c_4 \cdot (n - \ell)^2}{2} \right) \\
&= c_1 + c_2 \cdot n + \left(\sum_{\ell=1}^{n} \frac{(2 \cdot c_3 + c_4) \cdot \ell}{2} \right) + \left(\sum_{\ell=1}^{n} \frac{c_4 \cdot \ell^2}{2} \right) \\
&= c_1 + c_2 \cdot n + \frac{(2 \cdot c_3 + c_4)}{2} \cdot \sum_{\ell=1}^{n} \ell + \frac{c_4}{2} \cdot \sum_{\ell=1}^{n} \ell^2 \\
&= c_1 + c_2 \cdot n + \frac{(2 \cdot c_3 + c_4)}{2} \cdot \frac{n \cdot (n+1)}{2} + \frac{c_4}{2} \cdot \frac{n \cdot (n+1) \cdot (2 \cdot n + 1)}{6} \\
&\in \Theta(n^3)
\end{aligned}
$$

Maximum contiguous subsequence sum problem — Sol. 2

Problem: Find $\max \left\{ \sum_{i=\ell}^{h-1} a[i] \;\middle|\; 0 \le \ell \le h \le n \right\}$

where $a[0\,..\,n-1]$ is an array of integers (possibly negative) and $n \ge 0$.

Solution 2:

```
summax := 0;
for ℓ from 0 to n − 1 :
   for h from ℓ + 1 to n :                         sum := 0;
      {
         sum := 0;
         for i from l to h − 1 :                   sum := sum + a[h − 1];
            sum := sum + a[i];
         if summax < sum then summax := sum;
      }
```

Maximum contiguous subsequence sum problem — complexity of Sol. 2

Problem: Find $\max\left\{ \sum_{i=\ell}^{h-1} a[i] \;\middle|\; 0 \le \ell \le h \le n \right\}$
where $a[0\,..\,n-1]$ is an array of integers (possibly negative) and $n \ge 0$.

Solution 2:

```
summax := 0;
for ℓ from 0 to n − 1 :
  {
    sum := 0;
    for h from ℓ + 1 to n :
      {
        sum := sum + a[h − 1];
        if summax < sum then summax := sum;
      }
  }
```

Complexity:

$$c_1 + \sum_{\ell=0}^{n-1}\left(c_2 + \sum_{h=\ell+1}^{n} c_3\right) \;=\; c_1 + c_2\cdot n + \sum_{\ell=0}^{n-1} c_3\cdot(n-\ell) \;=\; c_1 + c_2\cdot n + c_3\cdot\sum_{\ell=1}^{n}\ell$$

$$\in\; c_1 + c_2\cdot n + c_3\cdot\frac{(n+1)\cdot n}{2} \;\in\; \Theta(n^2)$$

Maximum contiguous subsequence sum problem — Sol. 3

Problem: Find $\max\left\{ \sum_{i=\ell}^{h-1} a[i] \;\middle|\; 0 \le \ell \le h \le n \right\}$

where $a[0\,..\,n-1]$ is an array of integers (possibly negative) and $n \ge 0$.

Solution 3 (divide and conquer):

Either the maximum of the left half,
or the maximum of the right half,
or the max. of the left half touching the center
$+$ the max. of the right half touching the center

DIVIDE-AND-CONQUER method:

- split the problem into subproblems of roughly equal sizes,
- solve each subproblem recursively,
- do an extra small amount of work to combine the solutions of the subproblems into a solution of the problem.

```
maxsub(p, q) ≝
  if p + 1 = q then { if a[p] < 0 then 0 else a[p] }
  else {
        /* p + 2 ≤ q */
      k := ⌊(p + q)/2⌋ ; lmax := maxsub(p, k); rmax := maxsub(k, q);
      lbdmax := 0; lbd := 0; for i from k − 1 to p :
                          { lbd := lbd + a[i]; if lbdmax < lbd then lbdmax := lbd; }
      rbdmax := 0; rbd := 0; for i from k − 1 to p :
                          { rbd := rbd + a[i]; if rbdmax < rbd then rbdmax := rbd; }
      return max (lmax, rmax, lbdmax + rbdmax);
    }
```

Maximum contiguous subsequence sum problem — complexity of Sol. 3

Size of data: $n \stackrel{\text{def}}{=} q - p$

Recurrence:

$$t(n) \;=\; \begin{cases} c_1 & \text{if } n = 1 \\ 2 \cdot t(n/2) + c_2 \cdot n & \text{if } n > 1 \end{cases}$$

Complexity:

$$t(n) \; in \; \Theta(n \cdot \log n)$$

Theorem: (divide and conquer)
If t is defined by the recurrence

$$t(n) = a \cdot t(n/b) + c \cdot n^q \cdot (\log n)^k$$

where $a \geq 1$, $b > 1$, $c > 0$ and $k, q \geq 0$ then

$$t(n) \in \begin{cases} \Theta(n^{\log_b a}) & \text{if } a > b^q \\ \Theta(n^q \cdot (\log_b n)^{k+1}) & \text{if } a = b^q \\ \Theta(n^q \cdot (\log_b n)^k) & \text{if } a < b^q \end{cases}$$

Maximum contiguous subsequence sum problem — Sol. 4

Problem: Find $\max \left\{ \sum_{i=\ell}^{h-1} a[i] \,\middle|\, 0 \le \ell \le h \le n \right\}$

where $a[0\,..\,n-1]$ is an array of integers (possibly negative) and $n \ge 0$.

Solution 4:

```
summax := 0;  sum := 0;
for j from 0 to n − 1 :
  {
    sum := sum + a[j];
    if summax < sum then summax := sum;
    else if sum < 0 then sum := 0;
  }
```

Complexity: $\Theta(n)$

Warnings about constants

If we are only interested in asymptotic complexity, then:

constants in recurrences do not matter — **WRONG!**

$t_1(n) = 1 \cdot t_1(n-1) + 1$ — linear
$t_2(n) = 2 \cdot t_2(n-1) + 1$ — exponential
t_2 grows faster

constants in non-recursive formulas do not matter — **WRONG!**

$t_1(n) = 2^n$
$t_2(n) = 3^n$
t_2 grows faster

multiplying anything by a constants does not matter — **WRONG!**

$t_1(n) = 2^{1 \cdot n}$
$t_2(n) = 2^{2 \cdot n} y$
t_2 grows faster

adding anywhere a constant does not matter — **WRONG!**

$t_1(n) = (n+0)!$
$t_2(n) = (n+1)!$
t_2 grows faster

multiplying **the final value** by a constant or adding a constant to **the final value** does not matter — **RIGHT!**