Maximum sum of a subvector

We have a vector with $n$ locations $V[0 \ldots n-1]$. Write a program that reads a vector of integer numbers and writes the maximum sum of any contiguous subvector. For instance, if the input vector is the following:

```
2   ↓   6
↓    ↓
31  -41  59  26  -53  58  97  -93  -23  84
```

then the maximum sum is 187 and corresponds to $V[2 \ldots 6]$.

**Important:** assume that the vector is very long and you need an efficient algorithm.

**Question:** assume that the vector is so long that it cannot be fully stored in memory. Would it be possible to solve this problem by reading the input vector only once? Why or why not?

(From Jon Bentley, *Programming Pearls*, Addison-Wesley, 1986.)