

```
#include "question4.hpp"

using boost::numeric::ublas::matrix;

namespace question4 {

    matrix<int> allonsY(std::vector<std::tuple<int, int>> carnet,
                      int slotLength) {
        int spotCount(carnet.size()); // K

        matrix<int> dynaMat(spotCount+1, slotLength+1);

        for(int i = 0 ; i <= slotLength ; ++i) { // M[0,t] = 0
            dynaMat(0, i) = 0;
        }

        for(int i = 1 ; i <= spotCount ; ++i) { // M[i,j]

            int spotDuration, spotProfit;
            std::tie(spotDuration, spotProfit) = carnet[i-1]; // On met le gain du spot et sa durée dans spotProfit et spotDuration

            for(int j = 0 ; j <= slotLength ; j++) {
                if(spotDuration <= j) { // La longueur du spot est ≤ à celle restante
                    dynaMat(i,j) = std::max(dynaMat(i-1,j-spotDuration) + spotProfit,
                                             dynaMat(i-1,j)); // On prend le maximum entre M[k-1,t] et M[k-1,t-(durée du spot)] + (Gain du spot)
                } else { // La longueur du spot dépasse celle restante
                    dynaMat(i,j) = dynaMat(i-1,j);
                }
            }
        }

        return dynaMat;
    }
}
```