CSc 116 A02 - Fall 2018

Name:

Student Number:

On the back of the page, write a syntactically correct C++ function `sumLessThan` (including the function signature) that given a vector of integers and an integer n, will return the sum of all integers in the vector that are less than n. Your solution **may not contain loops** of any kind including for, while, and do-while loops (as well as `goto` statements). You may use the `forEach` function defined below or you may use recursion. Your implementation must work on all inputs, not just the ones provided below.

```
#include <iostream>
#include <functional>
#include <vector>

using Fn = std::function<void(int)>;

void forEach(std::vector<int> const& data, Fn f) {
  for (auto elem : data) {
      f(elem);
  }
}

// Your function here.
```

```
int main() {
  std::vector<int> data1 {};
  std::vector<int> data2 {
    1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

  auto sum1 = sumLessThan(data1, 10);
  auto sum2 = sumLessThan(data2, 10);

  std::cout << sum1 << ", ";
  std::cout << sum2 << std::endl;

  return 0;
}
```

When your code is correct, the output will be: `0, 45`

Consider the following possible solution:

```
int sumLessThan(std::vector<int> const& data, int n)
{
        int sum{};
    forEach(data,
            [&sum, n](int x)
            {
                sum += (x < n) ? x : 0;
            });
    return sum;
}
```

Here is an alternative solution using recursion:

```
int sumLessThanR(std::vector<int> data, int n)
{
    if (data.empty())
    {
        return 0;
    }

    int sum{};
    if (data.back() < n)
    {
        sum += data.back();
    }

    data.pop_back();
    return sum + sumLessThanR(data, n);
}
```