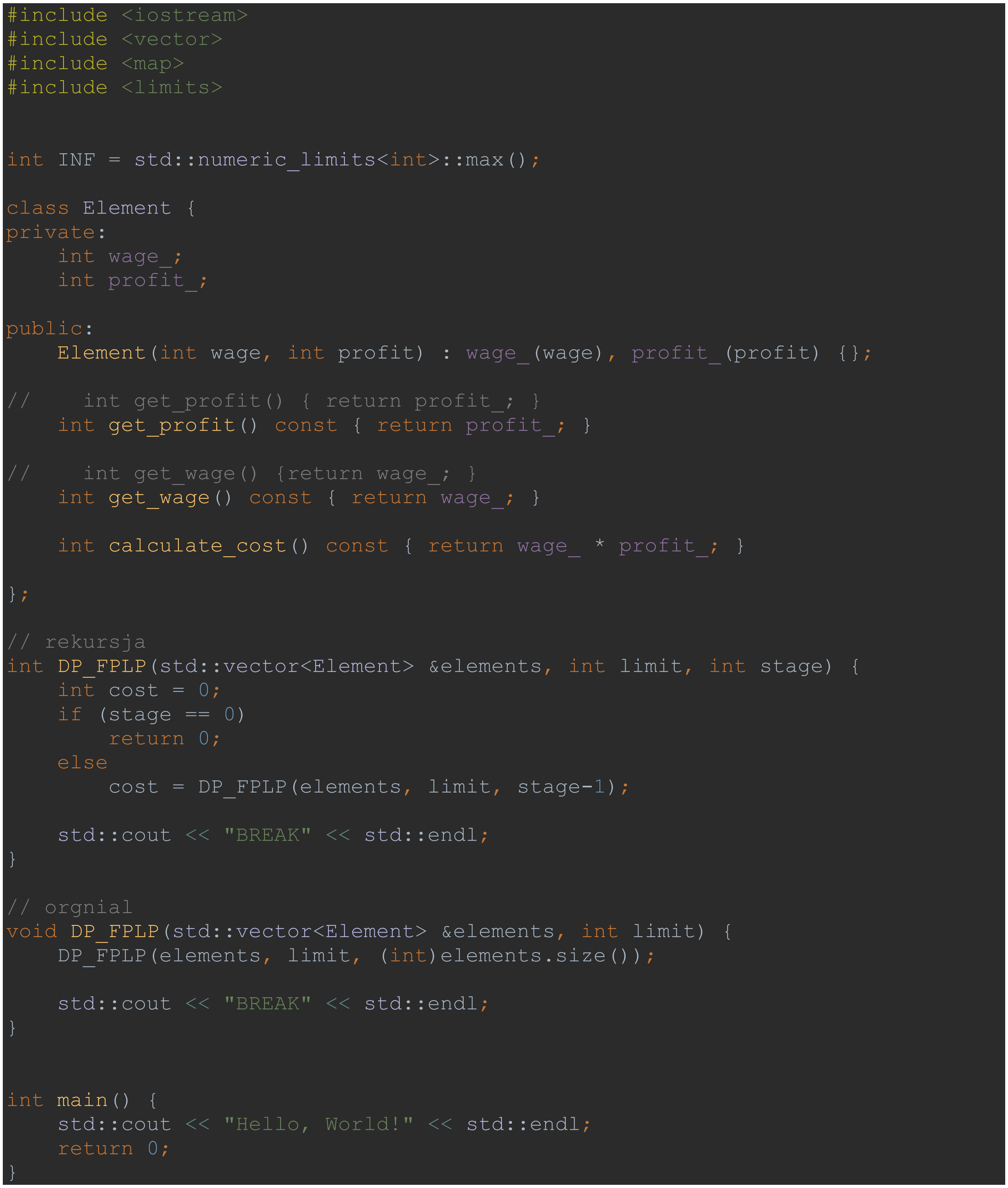

```
#include <iostream>
#include <vector>
#include <map>
#include <limits>


int INF = std::numeric_limits<int>::max();

class Element {
private:
    int wage_;
    int profit_;

public:
    Element(int wage, int profit) : wage_(wage), profit_(profit) {};

//    int get_profit() { return profit_; }
    int get_profit() const { return profit_; }

//    int get_wage() {return wage_; }
    int get_wage() const { return wage_; }

    int calculate_cost() const { return wage_ * profit_; }

};

// rekursja
int DP_FPLP(std::vector<Element> &elements, int limit, int stage) {
    int cost = 0;
    if (stage == 0)
        return 0;
    else
        cost = DP_FPLP(elements, limit, stage-1);

    std::cout << "BREAK" << std::endl;
}

// orgnial
void DP_FPLP(std::vector<Element> &elements, int limit) {
    DP_FPLP(elements, limit, (int)elements.size());

    std::cout << "BREAK" << std::endl;
}


int main() {
    std::cout << "Hello, World!" << std::endl;
    return 0;
}
```