

[Help](#)

```
#include <iostream>
#include <cmath>
#include <cstdlib>
#include <cstring>

using namespace std;

#include "
href../../../../common/math/ImportanceSampling_jl/src/MaxOption_h_src.pdfmath/Im
#include "
href../../../../common/math/jlparser/include/jlparser/parser_h_src.pdfjlparser/p

double MaxOption::payoff(const PnlMat *path_val)
{
    PnlVect S = pnl_vect_wrap_mat(path_val);
    double m = pnl_vect_max(&S);
    double sT = GET(&S, S.size - 1);
    return max(K * m - sT, 0.0);
}

MaxOption::MaxOption()
{
}

MaxOption::MaxOption(const Param &P)
    : BaseOption(P)
{
    label = "maximum";
    P.extract("strike", K);
}

void MaxOption::print() const
{
    cout << "**** Maximum Option Characteristics ****" << endl;
    BaseOption::print();
    cout << " strike : " << K << endl;
}
```

```
MaxOption::~MaxOption()  
{  
}
```