```
   Help
#if defined(PremiaCurrentVersion) && PremiaCurrentVersion <
     (2007+2) //The "#else" part of the code will be freely av
    ailable after the (year of creation of this file + 2)
#else

#include <vector>
#include <cmath>

#ifndef generator_h_
#define generator_h_

//random variable class
class rv
{
  //function simulates a random variable
 public:
  virtual double get_rv(void)=0;

  virtual ~rv() {};
};


//bernoulli random variable class
class rv_bernoulli: public rv
{

  //the parameters: probability(x=nvalue1)=nproba;   probab
    ility(x=nvalue2)=1-nproba
 private:
  double nproba;
  double nvalue1;
  double nvalue2;
  int generator;

 public:

  //class constructor
  rv_bernoulli(double _nproba=0.5, double _nvalue1=1,
    double _nvalue2=1,int _generator=1)
    {
```

```
      nproba=((_nproba>0.)&(_nproba<1.))?_nproba:0.5;
      generator=_generator;
      nvalue1=_nvalue1;
      nvalue2=_nvalue2;
    };

  //function simulates a bernoulli random variable
  virtual double get_rv(void)
  {
    double x;
    x=pnl_rand_uni(generator);
    return (x<nproba)? nvalue1: nvalue2;
  };

  virtual ~rv_bernoulli() {};
};

class rv_vector
{
  //parameters:
  //ndim_vector - a dimension of our vector
 protected:
  int ndim_vector;

 public:

  //class constructor
  rv_vector(int _ndim)
    { ndim_vector=(_ndim>0)? _ndim: 1;};

  virtual std::vector<double> get_rv(void)=0;
  virtual ~rv_vector() {};
};


#endif



#endif //PremiaCurrentVersion
```

References