NAME

readPolicyFile – Read policy data from a file

SYNOPSIS

#include <qnet.h>

int readPolicyFile(string *path***, int** *ndim***, int** *decn***, int** *trunc[]***, void*** *p***);**

DESCRIPTION

The **readPolicyFile**() reads policy information for the entire state space from the file whose name is *path*. The dimension of the state space must match *ndim*, the number of decisions for each state must be *decn*, and the state space truncations must match those in *trunc[]*. The policy data is returned in the memory pointed to by *p* which must have been allocated with one of the **makePolicyArray**(), **allocateArray**(), or **allocateArrayV**() functions.

RETURN VALUE

On success, the value **QNET_NORMAL** (defined to be zero) is returned, otherwise one of the following negative values is returned:

QNET_BAD_DIMENSION

The expected dimension of the state space (specified in *ndim*) does not match the dimension in the file.

QNET_BAD_DECISION

The expected number of decisions (specified in *decn*) does not match the number of decisions in the file.

QNET_BAD_TRUNCATION

At least one of the expected state space truncations (specified in *trunc[]*) does not match the corresponding truncation in the file.

QNET_BAD_FILE

The input file could not be opened or read from.

EXAMPLE

The following sequence can be used to read policy data from the file *policy.u* into the memory pointed to by *action*. First the header information is read

```
int maxdim = 3;    // largest state space dimension expected
int ndim;          // actual dimension of state space
int decn;          // number of decisions
int N[maxdim + 1]; // state space truncations

getPolicyFileInfo(string("policy.u"), maxdim, &ndim, &decn, &N[1]);
```

and the values of *ndim* and *decn* can be checked to make sure they match the expected values. Once this is done, the appropriately sized arrays can be declared and populated with policy data

```
int**** action = (int****) makePolicyArray(ndim, decn, &N[1]);
readPolicyFile(string("policy.u"), ndim, decn, &N[1], action);
```

Note that N[0] is not used; this follows the convention used in most of the QNET DP code.

SEE ALSO

getPolicyFileInfo(3), **writePolicyFile**(3), **makePolicyArray**(3)

AUTHOR

Copyright © 2007-2008 Jonathan R. Senning, Department of Mathematics and Computer Science, Gordon College, 255 Grapevine Road, Wenham MA, 01984.