Filename: minerprocess.cpp

Header:

#include "urlapi.h"

#include "structures.h"

#include "jsonparser.h"

#include <QObject>

#include <QProcess>

#include <QTextEdit>

#include <QThread>

#include <QSettings>

Class name: PoolInfoThread

Inherits: QThread

Description: PoolInfoThread is a class that periodically asks the parent object (of type MinerProcess) to update the pool information. The parent object then uses API provided by mining pools to fetch real-time mining pool status such as daily output, latency, and price of cryptos.

Constructor:

**PoolInfoThread**(float refresh\_rate, QObject\* pParent = Q\_NULLPTR):

Parameters: Refresh rate of pool info (seconds); Pointer to the parent object.

Return: None.

Task: Constructs a PoolInfoThrd object.

Key Protected Fields:

|  |  |
| --- | --- |
| float refresh\_rate = 60 | The refresh rate (seconds) of pool information. |
| MinerProcess\* \_pParent | The parent object of this object. |

Public Methods:

void ***run***():

Parameters: None.

Return: None.

Task: Periodically ask the parent object to update pool information.

Class name: MiningInfoThread

Inherits: QThread

Description: MiningInfoThread is a class that periodically asks the parent object (of type MinerProcess) to update the mining information. The parent object then uses API provided by mining core to fetch real-time mining status such as hash rate and accepted shares.

Constructor:

**MiningInfoThread**(float refresh\_rate, QObject\* pParent = Q\_NULLPTR):

Parameters: Refresh rate of mining info (seconds); Pointer to the parent object.

Return: None.

Task: Constructs a MiningInfoThrd object.

Key Protected Fields:

|  |  |
| --- | --- |
| float refresh\_rate = 3 | The refresh rate (seconds) of mining information. |
| MinerProcess\* \_pParent | The parent object of this object. |

Public Methods:

void ***run***():

Parameters: None.

Return: None.

Task: Periodically ask the parent object to update mining information.

Class name: MinerProcess

Inherits: QObject

Description: MinerProcess runs the mining core. Then, it starts a thread (PoolInfoThrd) to periodically fetch the mining pool information and a thread (MiningInfoThrd) to periodically fetch the mining information.

Constructor:

**NvMonitorThrd**(QObject\* p = Q\_NULLPTR, NvidiaAPI \*nvapi = NULL):

Parameters: Pointer to the parent object; Pointer to the NVIDIA API.

Return: None.

Task: Constructs a NvMonitorThrd object.

Key Private Fields:

|  |  |
| --- | --- |
| UrlAPI\* urlAPI | The API to fetch data from a URL. |
| std::string core\_api\_str | The API URL of the mining core. |
| MinerJsonParser\* jsonParser = nullptr | Pointer to a MinerJsonParser. |
| std::string pool\_api\_str | The API URL of the mining pool. |
| PoolJsonParser\* poolJsonParser = nullptr | Pointer to a PoolJsonParser. |
| MainWindow\* mainWindow | Pointer to the MainWindow object. |
| QProcess \_miner | The process that runs the mining core. |
| PoolInfoThread\* \_poolInfoThread | The thread that periodically updates pool info. |
| MiningInfoThread\* \_miningInfoThread | The thread that periodically updates mining info. |
| QTextEdit\* \_log | The TextEdit object for log. |
| QString \_minerPath | The path of the mining core. |
| QString \_minerArgs | The arguments for the mining core. |
| QSettings\* \_settings | Pointer to the QSettings object. |
| QString \_outHelper = QString() | Temporary string for log output. |
| bool \_isRunning | Bool variable indicates whether the core is running. |

Private Methods:

void **start**(const QString& path, const QString& args):

Parameters: Path of the mining core; Arguments to the mining core.

Return: None.

Task: Start a process to run the mining core.

void **stop**():

Parameters: None.

Return: None.

Task: Stop the mining core.

void **SetAPI**(Core\* core):

Parameters: None.

Return: None.

Task: Get and record the API of the mining core.

MiningInfo **getStatus**():

Parameters: None.

Return: Information about mining.

Task: Fetch and pack the information about mining.

QList<PoolInfo> **getPoolStatus**():

Parameters: None.

Return: A list of information about mining pool.

Task: Fetch and pack the information about mining pool.

QList<unsigned long> **GetChildrenPID**(unsigned long ppid):

Parameters: PID of the parent process.

Return: A list of children’s pids.

Task: Fetch the children processes’ pids.

void **refreshMiningInfo**():

Parameters: None..

Return: None.

Task: Get and emit the mining information.

void **refreshPoolInfo**():

Parameters: None..

Return: None.

Task: Get and emit the pool information.

Public Slots:

void **onReadyToReadStdout**():

Parameters: None.

Return: None.

Task: Read the standard output of the mining core.

void **onExit**():

Parameters: None.

Return: None.

Task: Emit the stop signal.

void **onStarted**():

Parameters: None.

Return: None.

Task: Emit the start signal.

Signals:

void **emitStarted**():

Parameters: None.

Return: None.

Task: Indicate the mining core has started.

void **emitStopped**():

Parameters: None.

Return: None.

Task: Indicate the mining core has stopped.

void **emitError**():

Parameters: None.

Return: None.

Task: Indicate the mining core generates an error.

void **emitMiningInfo**(MiningInfo mingInfo):

Parameters: None.

Return: None.

Task: Carry the mining information.

void **emitPoolInfo**(QList<PoolInfo> poolInfos):

Parameters: None.

Return: None.

Task: Carry the pool information.