class Idea

Idea::Idea(string propos, string cont)

Notation: BigO(1)

int Idea::getId( int id ) const

Complexity: 1

Notation: BigO(1)

string Idea::getProper( ) const

Complexity: 1

Notation: BigO(1)

string Idea::getContent( ) const

Complexity: 1

Notation: BigO(1)

vector< string > Idea::getKeyWord( ) const

Complexity: 1

Notation: BigO(1)

void Idea::setProper( string p )

Complexity: 1

Notation: BigO(1)

void Idea::setContent( string c )

Complexity: 1

Notation: BigO(1)

void Idea::addKeyWord( string keyword )

Complexity: 1

Notation: BigO(1)

**bool Idea::checkKeyWordVector( string word )**

Complexity: 3n+3 [worst case]

Notation: BigO(n)

void Idea::printIdeaContent()

Complexity: 6n+13 [worst case]

Notation: BigO(n)

void Idea::addKeyWordToVector( vector< string >& vecWords )

Complexity: 3n^2+ 6n + 2 [worst case]

Notation: BigO(n^2)

void Idea::addContentTokenizeToVect(vector<string>& vecWords)

Complexity: 4n^2 + 8n + 3 [worst case]

Notation: BigO(n^2)

bool Idea::checkKeyContentsIdeaSearch( string word )

Complexity: n^2 + 2n + 1 [worst case]

Notation: BigO(n^2)

bool Idea::checkContent( string word )

Complexity: n + 4n [worst case]

Notation: BigO(n)

bool Idea::operator ==(Idea id) const

Complexity: 2 [worst case]

Notation: BigO(1)