コーディングルール

作成者：杉山涼

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* !@file ファイル名を記載

\* @brief ファイル概要もしくはクラス概要

\* @author 作成者名

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//クラス名はアッパーキャメル

class ClassName {

//アクセス指定子は

private:

public:

protected:

//の順に記載すること

//メンバ定数はアッパーキャメル

//接頭にはmcを付けること

const int mcConstInt;

//静的メンバ変数はアッパーキャメル

//接頭にはmsをつけること

static int msStaticint;

//メンバ変数はロワーキャメル

//接頭には必ずm\_をつけること

int m\_privateInt;

//変数のコメントのつけ方

int m\_protectedInt //@brief 変数の説明

//メンバ関数はアッパーキャメル

//接頭にはmを付けること

void mPrivateFunction();

//引数の接頭には\_を付けること

void mPublicFunction(int \_pram);

//関数のコメントのつけ方

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* @brief 関数の概要

\* @param[in] 引数の説明

\* @return 戻り値の説明

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

};

//インターフェイスクラス

class IclassName　{

};

//基本的に、コメントを振らなくてもわかるような名前の付け方を心掛けること