#pragma once

#include <twml/defines.h>

#include <cstddef>

#include <vector>

#include <string>

#ifdef \_\_cplusplus

extern "C" {

#endif

struct twml\_tensor\_\_;

typedef twml\_tensor\_\_ \* twml\_tensor;

#ifdef \_\_cplusplus

}

#endif

#ifdef \_\_cplusplus

namespace twml {

class TWMLAPI Tensor

{

private:

twml\_type m\_type;

void \*m\_data;

std::vector<uint64\_t> m\_dims;

std::vector<uint64\_t> m\_strides;

public:

Tensor() {}

Tensor(void \*data, int ndims, const uint64\_t \*dims, const uint64\_t \*strides, twml\_type type);

Tensor(void \*data, const std::vector<uint64\_t> &dims, const std::vector<uint64\_t> &strides, twml\_type type);

const std::vector<uint64\_t>& getDims() const {

return m\_dims;

}

int getNumDims() const;

uint64\_t getDim(int dim) const;

uint64\_t getStride(int dim) const;

uint64\_t getNumElements() const;

twml\_type getType() const;

twml\_tensor getHandle();

const twml\_tensor getHandle() const;

template<typename T> T \*getData();

template<typename T> const T \*getData() const;

};

TWMLAPI std::string getTypeName(twml\_type type);

TWMLAPI const Tensor \*getConstTensor(const twml\_tensor t);

TWMLAPI Tensor \*getTensor(twml\_tensor t);

TWMLAPI uint64\_t getSizeOf(twml\_type type);

}

#endif

#ifdef \_\_cplusplus

extern "C" {

#endif

TWMLAPI twml\_err twml\_tensor\_create(twml\_tensor \*tensor, void \*data,

int ndims, uint64\_t \*dims,

uint64\_t \*strides, twml\_type type);

TWMLAPI twml\_err twml\_tensor\_delete(const twml\_tensor tensor);

TWMLAPI twml\_err twml\_tensor\_get\_type(twml\_type \*type, const twml\_tensor tensor);

TWMLAPI twml\_err twml\_tensor\_get\_data(void \*\*data, const twml\_tensor tensor);

TWMLAPI twml\_err twml\_tensor\_get\_dim(uint64\_t \*dim, const twml\_tensor tensor, int id);

TWMLAPI twml\_err twml\_tensor\_get\_num\_dims(int \*ndims, const twml\_tensor tensor);

TWMLAPI twml\_err twml\_tensor\_get\_num\_elements(uint64\_t \*nelements, const twml\_tensor tensor);

TWMLAPI twml\_err twml\_tensor\_get\_stride(uint64\_t \*stride, const twml\_tensor tensor, int id);

#ifdef \_\_cplusplus

}

#endif