

Learning the basics QUIZ with ANSWERS

Section 2: HDF File Organization

1. Name and describe the two primary objects that can be stored in an HDF5 file.

Answers: Group: A grouping structure containing zero or more HDF5 objects, together with supporting metadata. Dataset: A multidimensional array of data elements, together with supporting metadata.

2. What is an attribute?

Answer: An HDF5 attribute is a user-defined HDF5 structure that provides extra information about an HDF5 object.

3. Give the path name for an object called `harry` that is a member of a group called `dick`, which, in turn, is a member of the root group.

Answer: `/dick/harry`

Section 3: The HDF5 API

1. Describe the purpose of each of the following HDF5 APIs:

H5A, H5D, H5E, H5F, H5G, H5T, H5Z

Answers:

H5A: Attribute access and manipulation routines

H5D: Dataset access and manipulation routines

H5E: Error handling routines H5F: File access routines

H5G: Routines for creating and operating on groups

H5T: Routines for creating and manipulating the datatypes of dataset elements

H5Z: Data compression routines

Section 4: Creating an HDF5 File

1. What two HDF5 routines must be called to create an HDF5 file?

Answer: `H5Fcreate` and `H5Fclose`.

2. What include file must be included in any file that uses the HDF5 library?

Answer: `hdf5.h` must be included because it contains definitions and declarations used by the library.

3. An HDF5 file is never completely empty because as soon as it is created, it automatically contains a certain primary object. What is that object?

Answer: The root group.

Section 5: Creating a Dataset

1. Name and describe two major datatype categories.

Answers:

Atomic datatype: An atomic datatype cannot be decomposed into smaller units at the API level.

Compound datatype: A compound datatype is a collection of atomic and compound datatypes, or small arrays of such types.

2. List the HDF5 atomic datatypes. Give an example of a predefined datatype. How would you create a string dataset?

Answers:

There are six HDF5 atomic datatypes: *integer, floating point, date and time, character string, bit field, and opaque.*

Examples of predefined datatypes include the following:

`H5T_IEEE_F32LE` - 4-byte little-endian, IEEE floating point

H5T_NATIVE_INT - native integer

You would create a string dataset with the H5T_C_S1 datatype, and set the size of the string with the H5Tset_size call.

3. What does the dataspace describe? What are the major characteristics of the simple dataspace?

Answers: The dataspace describes the dimensionality of the dataset. A simple dataspace is characterized by its rank and dimension sizes.

4. What information needs to be passed to the H5Dcreate function, i.e., what information is needed to describe a dataset at creation time?

Answer: The dataset location, name, dataspace, datatype, and dataset creation property list.

Section 6: Reading from and Writing to a Dataset

1. What are six pieces of information which need to be specified for reading and writing a dataset?

Answer: The dataset identifier, the dataset's datatype and dataspace in memory, the dataspace in the file, the dataset transfer property list, and a data buffer.

2. Why are both the memory dataspace and file dataspace needed for read/write operations, while only the memory datatype is required?

Answer: A dataset's file datatype is not required for a read/write operation because the file datatype is specified when the dataset is created and cannot be changed. Both file and memory dataspace are required for dataset subsetting and for performing partial I/O operations.

3. In Figure 6.1, what does this line mean?:

```
DATASPACE { SIMPLE ( 4 , 6 ) / ( 4 , 6 ) }
```

Answer: It means that the dataset dset has a simple dataspace with the current dimensions (4,6) and the maximum size of the dimensions (4,6).

Section 7: Creating an Attribute

1. What is an attribute?

Answer: An attribute is a dataset attached to an object. It describes the nature and/or the intended usage of the object.

2. Can partial I/O operations be performed on attributes?

Answer: No.

Section 8: Creating a Group

1. What are the two primary objects that can be included in a group?

Answer: A group and a dataset.

Section 9: Creating Groups Using Absolute and Relative Names

1. Group names can be specified in two ways. What are these two types of group names?

Answer: Relative and absolute.

2. You have a dataset named moo in the group boo, which is in the group foo, which, in turn, is in the root group. How would you specify an absolute name to access this dataset?

Answer: /foo/boo/moo

Section 10: Creating Datasets in Groups

1. Describe a way to access the dataset moo described in the previous section (Section 9, question 2) using a relative name. Describe a way to access the same dataset using an absolute name.

Answers:

Access the group /foo and get the group ID. Access the group boo using the group ID obtained in Step 1. Access the dataset moo using the group ID obtained in Step 2.

```
gid = H5Gopen (file_id, "/foo", 0); /* absolute path */  
gid1 = H5Gopen (gid, "boo", 0); /* relative path */  
did = H5Dopen (gid1, "moo"); /* relative path */
```

Access the group /foo and get the group ID. Access the dataset boo/moo with the group ID just obtained.

```
gid = H5Gopen (file_id, "/foo", 0); /* absolute path */  
did = H5Dopen (gid, "boo/moo"); /* relative path */
```

Access the dataset with an absolute path.

```
did = H5Dopen (file_id, "/foo/boo/moo"); /* absolute path */
```