

Microsoft 70-691

TS: Windows HPC Server 2008, Developing

Version: 4.0

QUESTION NO: 1

You plan to develop a parallel application by using Windows HPC Server 2008. You plan to use a method to control the calculations performed by each individual process. You need to ensure that different commands can be sent to each process. Which function should you use?

- A. MPI_Send
- B. MPI_Bcast
- C. MPI_Gather
- D. MPI_Reduce

Answer: A

Explanation:

QUESTION NO: 2

You plan to develop a parallel application by using Windows HPC Server 2008. You need to implement a method that will send different data to each process in a communicator by using a single call. Which function should you use?

- A. MPI_Send
- B. MPI_Gather
- C. MPI_Reduce
- D. MPI_Scatter

Answer: D

Explanation:

QUESTION NO: 3

You plan to develop a parallel application by using Windows HPC Server 2008. The application performs the following tasks: -Performs multistep calculations -Processes data by using the same code segment that runs on each compute node You need to ensure that each compute process executes the same step simultaneously. Which type of communication should you use?

- A. Buffered
- B. Collective
- C. Non-blocking
- D. Point-to-point

Answer: B

Explanation:

QUESTION NO: 4

You plan to develop a parallel application by using Windows HPC Server 2008. You need to ensure that when the application starts, a parameter value is communicated to each process in the cluster by using a single call. Which function should you use?

- A. MPI_Send
- B. MPI_Bcast
- C. MPI_Gather
- D. MPI_Reduce

Answer: B

Explanation:

QUESTION NO: 5

You plan to develop a single program multiple data (SPMD) application by using Windows HPC Server 2008. You use multiple processes to perform intermediate calculations and to provide the results as a single number. You need to ensure that the intermediate results are collected and added together by using a single MPI function. Which function should you use?

- A. MPI_Send
- B. MPI_Bcast
- C. MPI_Gather
- D. MPI_Reduce

Answer: D

Explanation:

QUESTION NO: 6

You develop a parallel application that will be deployed on a Windows HPC Server 2008 cluster. You write the following code segment. (Line numbers are included for reference only.)

```
01 int rank
02 MPI_Comm_rank(MPI_COMM_WORLD, &rank)
03 // declaration of variable hostName
04 // code to execute
05
06 std::cout << "hostname of rank " << rank
```

```
07 std::cout << ": " << hostName << "\n"
```

You need to ensure that a variable named `hostName` contains the name of the cluster node that the code runs on. Which code segment should you add at line 05?

- A. `char *hostName = getenv("MPI_HOSTNAME")`
- B. `char hostName[MPI_MAX_PROCESSOR_NAME] int resultlen
MPI_Get_processor_name(&resultlen, hostName)`
- C. `char hostName[MPI_MAX_NAME_STRING] int resultlen
MPI_Comm_get_name(MPI_COMM_WORLD, hostName, &resultlen)`
- D. `char hostName[256] int resultlen = sizeof(hostName) if (rank == 0) { gethostname(hostName, resultlen) }`

Answer: B

Explanation:

QUESTION NO: 7

You develop a parallel application that will be deployed on a Windows HPC Server 2008 cluster. You write the following code segment. (Line numbers are included for reference only.) 01 `MPI_Init(&argc, &argv)` 02 03 { 04 // program part for rank 0 05 06 } 06 else 07 { 08 // program part for all other ranks 09 } 10 `MPI_Finalize()` You need to ensure that the code in the rank 0 section only executes on the process that has a rank of 0. Which code segment should you insert at line 02?

- A. `int size MPI_Comm_size(MPI_COMM_WORLD, &size) if (size > 0)`
- B. `int size MPI_Comm_size(MPI_COMM_WORLD, &size) if (size == 0)`
- C. `int rank MPI_Comm_rank(MPI_COMM_WORLD, &rank)if (rank == 0)`
- D. `char hostName[MPI_MAX_PROCESSOR_NAME] int resultlen
MPI_Get_processor_name(hostName, &resultlen) char masterName[] = "rank0" if
(strcmp(masterName, hostName) != 0)`

Answer: C

Explanation:

QUESTION NO: 8

You develop a parallel application that will be deployed on a Windows HPC Server 2008 cluster. The application uses Point-to-point communication. The process that has a rank 0 sends an array of integer values to the rank 1 process by using the following code segment. (Line numbers are included for reference only.) 01 `int values[5]` 02 `MPI_Send(values, 5, MPI_INT, 1, 42,` 03 `MPI_COMM_WORLD)` You need to ensure that each receiving process is able to correctly receive the values in the most efficient manner possible. Which code segment should you use?

- A.** int values[5] int buffer[5] MPI_Reduce(buffer, values, 5, MPI_INT, MPI_SUM, 1, MPI_COMM_WORLD)
- B.** int values[5] MPI_Status status MPI_Recv(values, 5, MPI_INT, 0, 42, MPI_COMM_WORLD, &status) C. int value MPI_Status status MPI_Recv(&value, 1, MPI_INT, 0, 42, MPI_COMM_WORLD, &status)
- C.** int values[5] MPI_Status status MPI_Recv(values, sizeof(int), MPI_INT, 0, 42, MPI_COMM_WORLD, &status)

Answer: B

Explanation:

QUESTION NO: 9

You develop a parallel application that will be deployed on a Windows HPC Server 2008 cluster. You write the following code segment that computes an approximation to Pi. (Line numbers are included for reference only.)

```
01 static int totDarts = 10000
```

```
02 int numDarts = 0
```

```
03
```

```
04 MPI_Comm_size(MPI_COMM_WORLD, &commSize)
```

```
05 MPI_Comm_rank(MPI_COMM_WORLD, &commRank)
```

```
06
```

```
07 for(i = 0; i < totDarts; i++) {
```

```
08 x = 2 * (randomNr() - 0.5)
```

```
09 y = 2 * (randomNr() - 0.5)
```

```
10 if (x*x + y*y <= 1.0)
```

```
11 ++numDarts
```

```
12 }
```

```
13
```

```
14 if(commRank == 0) {
```

```
15 pi = 4 * numDarts / (commSize * totDarts)
```

```
16 printf("Pi = %d\n", pi)
```

```
17 }
```

You need to ensure that the total number of numDarts is only available on the process that

has a rank 0. Which code segment should you insert at line 13?

- A. `MPI_Reduce(&numDarts, &total, 1, MPI_INT, MPI_SUM, 0, MPI_COMM_WORLD)`
- B. `MPI_Allreduce(&total, &numDarts, 1, MPI_INT, MPI_SUM, MPI_COMM_WORLD)`
- C. `MPI_Gather(&numDarts, 1, MPI_INT, &total, 1, MPI_INT, 0, MPI_COMM_WORLD)`
- D. `MPI_Allgather(&numDarts, 1, MPI_INT, &total, 1, MPI_INT, MPI_COMM_WORLD)`

Answer: A

Explanation:

QUESTION NO: 10

You plan to develop a Microsoft Message Passing Interface (MPI) application by using Microsoft Visual Studio. You have installed the Microsoft HPC Pack 2008 SDK at C:\Program Files\Microsoft HPC Pack 2008 SDK. You create a new console application. You need to ensure that the application compiles for a 64bit architecture. Which two tasks should you perform? (Each correct answer presents part of the solution. Choose two.)

- A. Set the Import Library property to `msmpi.lib`.
- B. Set the Additional Dependencies property to `msmpi.lib`.
- C. Set the Additional Library Directories property to `C:\Program Files\Microsoft HPC Pack 2008 SDK\lib\amd64`.
- D. Set the Additional Manifest Dependencies property to `C:\Program Files\Microsoft HPC Pack 2008 SDK\lib\amd64`.

Answer: B,C

Explanation:

QUESTION NO: 11

You develop a parallel application by using Microsoft Visual Studio 2008. You have installed the Microsoft HPC Pack 2008 SDK at C:\Program Files\Microsoft HPC Pack 2008 SDK. You create a new console application based on Microsoft Message Passing Interface (MPI). You set the Debugger to Launch property to MPI Cluster Debugger. You need to ensure that five processes are launched when the application is run in debug mode in Visual Studio 2008. Which two tasks should you perform? (Each correct answer presents part of the solution. Choose two.)

- A. Set the MPIRun Arguments property to `n 5`.
- B. Set the Application Arguments property to `n 5`.
- C. Set the MPIRun Working Directory property to `C:\Program Files\Microsoft HPC Pack 2008 SDK\bin`.
- D. Set the MPIRun Command property to `C:\Program Files\Microsoft HPC Pack 2008 SDK\bin\MPIExec`. E. Set the Application Command property to `C:\Program Files\Microsoft HPC`

Pack 2008 SDK\bin\MPIExec.

Answer: A,D

Explanation:

QUESTION NO: 12

You create a parallel application by using Microsoft Visual Studio 2008. The application prints a message to the console of all Microsoft Message Passing Interface (MPI) processes except for one. The application terminates and displays an error message. You need to change the code in the application so that the application completes successfully. Which set of tasks should you include in the application?

- A. Include mpi.h Link to msmpi.lib Call MPI_Start Call MPI_Comm_rank Call MPI_Finalize
- B. Include mpi.h Link to msmpi.lib Call MPI_Init Call MPI_Comm_rank Call MPI_Finalize
- C. Include mpi.h Link to ccpapi.tlb Call MPI_Init Call MPI_Comm_rank
- D. Include mpi.h Link to msmpi.lib Call MPI_Init Call MPI_Comm_size

Answer: B

Explanation:

QUESTION NO: 13

You plan to create a parallel application by using Microsoft Visual C++. You want to use the Microsoft Message Passing Interface (MPI) library in the application. You need to ensure that the application can be compiled for both 32bit and 64bit platforms. You also need to ensure that targeting both platforms requires minimum maintenance. What should you do?

- A. Create a separate project for each platform.
- B. Create separate solution files for each platform.
- C. Create separate custom build rules for each platform.
- D. Create a separate build configuration for each platform.

Answer: D

Explanation:

QUESTION NO: 14

You create a Microsoft Message Passing Interface (MPI) application by using Microsoft Visual Studio 2008. The application process is named MPIApplication.exe. You plan to debug the application by using the MPI Cluster Debugger in Visual Studio 2008. You need to attach the debugger to the remote process on a compute node. Which process should you attach to?

- A. smpd.exe
- B. mpiexec.exe
- C. msvmon.exe
- D. MPIApplication.exe

Answer: C

Explanation:

QUESTION NO: 15

You create a Microsoft Message Passing Interface (MPI) application by using Microsoft Visual Studio 2008. You plan to debug the application by using Visual Studio 2008. You need to configure the nodes that will be used for debugging. What should you do?

- A. Use the Cluster Debugger Configuration tool in Visual Studio.
- B. Use the debugging pane in the Visual Studio project property pages.
- C. Use the configuration pane in the Cluster Administration Console tool.
- D. Use the node management pane in the Cluster Administration Console tool.

Answer: A

Explanation:

QUESTION NO: 16

You are an application developer. You plan to debug a Microsoft Message Parsing Interface (MPI) application by using the MPI Cluster Debugger. You set appropriate breakpoints in the application. You need to evaluate the value of a variable on different MPI ranks when a breakpoint is encountered. What should you do?

- A. Place the mouse pointer on top of the variable.
- B. Add the variable to the Watch window. Use the Modules window to switch between MPI ranks.
- C. Add the variable to the Watch window. Use the Threads window to switch between MPI ranks.
- D. Add the variable to the Watch window. Use the Processes window to switch between MPI ranks.

Answer: D

Explanation:

QUESTION NO: 17

You create a new Microsoft Message Passing Interface (MPI) application. The application will be deployed on a Windows HPC Server 2008 cluster. You plan to debug the application by using the

MPI Cluster Debugger in Visual Studio 2008. You need to ensure that you can successfully launch and use the MPI Cluster Debugger. What should you do?

- A. Run a broker node on the cluster.
- B. Assign your user account to the Administrator role on the cluster.
- C. In the MPI Cluster Debugger, set the Deploy Common Runtime property to Yes.
- D. Connect the head node to the enterprise network, a private network, and a separate application network. Connect the compute nodes only to the private network and the application network.

Answer: C

Explanation:

QUESTION NO: 18

You review the design of a module of an online banking system. The module will provide real-time statistics related to payments made by the users. The module must support 1,000 concurrent users. A new instance named Thread1 is created for every user who accesses the banking system. The following pseudocode represents the design of the module. (Line numbers are included for reference only.)

01 var amount = 0 // shared variable represents amount of transfer

02 var credit = 0 // shared variable represents total credit

03 var debit = 0 // shared variable represents total debit

04 Thread1:

05 reads userInput from console

06 amount = userInput

07 creates new Thread2 instance and runs it

08 creates new Thread3 instance and runs it

09 Thread2:

10 value = amount

11 if (value > 0) debit = debit + value

12 Thread3:

13 value = amount

14