Calculate the number π using following formulas or other (any which you like):

$$\pi = 2\sqrt{3} \sum_{k=0}^{\infty} \frac{(-1)^k}{3^k (2k+1)}$$

$$\pi = \sum_{k=0}^{\infty} \frac{(-1)^k}{4^k} \left(\frac{2}{4k+1} + \frac{2}{4k+2} + \frac{1}{4k+3} \right)$$

Use only **send** and **receive** commands to exchange the messages between processors.

If a process calls Recv it will simply wait until a message from the corresponding Send is received before proceeding. Similarly the Send will wait until the message has been revieved by the corresponding Recv.

Here are the actual definitions of the Send and Recv methods:

comm.send(buf, dest=0, tag=0)

Performs a basic send. This send is a point-to-point communication. It sends information from exactly one process to exactly one other process.

comm (MPI comm) – communicator we wish to query

buf (choice) – data to send

dest (integer) – rank of destination

tag (integer) – message tag

comm.recv(buf, source=0, tag=0)

Performs a point-to-point receive of data.

comm (MPI comm) – communicator we wish to query

buf (choice) – initial address of receive buffer (choose receipt location)

source (integer) – rank of source

tag (integer) – message tag

Sometimes there are cases when a process might have to send many different types of messages to another process. Instead of having to go through extra measures to

differentiate all these messages, MPI allows senders and receivers to also specify message IDs (known as tags) with the message. The receiving process can then request a message with a certain tag number and messages with different tags will be buffered until the process requests them.