

Przetwarzanie współbieżne

MPI

Cel:

- Opanowanie modelu przesyłania komunikatów MPI

Kroki:

1. Utworzenie katalogu roboczego (np. lab_10)
2. Przegranie przykładowego [programu](#), jego kompilacja (mpicc) i uruchomienie (mpirun).

np.

```
mpicc przykład.c -o prz_mpi
```

```
mpirun -np 7 ./prz_mpi
```

3. Napisz program, który po sczytaniu wpisanej liczby rozpropaguje ją w konwencji pierścienia. Proces numer i powinien dostawać liczbę i przesyłać do procesu $i+1$ dopóki ostatni proces nie zostanie osiągnięty. Wartości powinny być sczytywane dopóki podana wartość nie jest ujemna.

Przykładowy wydruk:

Podaj liczbę:

5

Proces 1 dostal 5 od procesu 0

Proces 2 dostal 5 od procesu 1

...

Proces 20 dostal 5 od procesu 19

Proces 0 dostal 5 od procesu 20

Podaj liczbę:

-1

4. Napisz program sczytujący wpisana liczbę i propagujący ją na wszystkie węzły (MPI_Bcast). Każdy proces musi wypisywać swój identyfikator oraz wartość otrzymanej liczby. Wartości powinny być sczytywane dopóki podana wartość nie jest ujemna.

Warunki zaliczenia:

1. Obecność na zajęciach i wykonanie wszystkich kroków.
2. Oddanie sprawozdania z opisem zadania, kodem źródłowym programów, wynikami i wnioskami. Osobno należy załączyć cały kod źródłowy z komentarzami - w sprawozdaniu umieszczać tylko istotne fragmenty kodu!

Informacje konfiguracyjne:

- Istnieje kilka implementacji MPI i wymaga on narzędzi do kompilacji i uruchamiania:
 - W systemach Linux istnieją dwie darmowe implementacje MPI – mpiCH oraz OpenMPI – w celu obsługi tego drugiego potrzebne jest zainstalowanie pakietów **openmpi-bin** oraz **libopenmpi-dev** dla systemów opartych na dystrybucji Debian – (np. Ubuntu) lub **openmpi** oraz **openmpi-devel** dla systemów opartych na dystrybucji Red Hat (np. Fedora)
 - W systemie Windows istnieje wersja Microsoft MPI – najczęściej instalowana w pakiecie z Visual Studio ale dostępna też do samodzielnej instalacji
<https://docs.microsoft.com/en-us/message-passing-interface/microsoft-mpi>