МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Навчально-науковий Інститут

Комп’ютерних Інформаційних технологій

Кафедра комп’ютеризованих систем управління

Лабораторна робота№5

З дисципліни “Паралельні та розподілені обчислення”

На тему “ **Програмування для комп’ютерних систем зі локальною пам’яттю**”

Виконав:

Студент ННІКІТ СП-424

Єлістраток К.Є

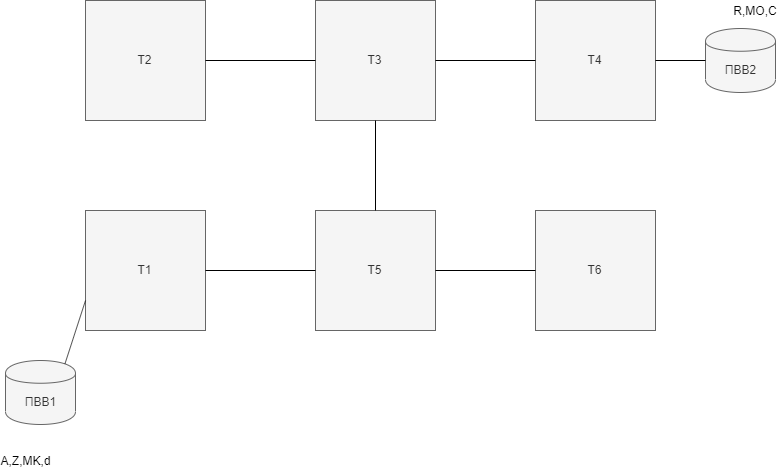
Перевірив:

Корочкін О В

Київ 2019

**ЗАВДАННЯ НА РОБОТУ**

**1. Структура паралельної комп’ютерної системи**



1. **Математична задача**

A = min(Z) \* R + d \* C \* (MO \* MK)

1. **Мова програмування:** PYTHON
2. **Засоби програмування взаємодії потоків (задач)** : MPI

**ВИКОНАННЯ РОБОТИ**

**Етап 1. Побудова паралельного алгоритму**

1. xi = min(Zh), i = 1,6
2. x = min(x,xi), i = 1,6
3. MUh = (MOh \* MK)
4. Ah = x \* Rh + d \* C \* MUh

**Етап 2. Розроблення алгоритмів потоків (задач) .**

|  |  |
| --- | --- |
| **№** | **Задача *Т1*** |
| **1** | Введення Z,MK,d |
| **2** | x = min(Zh) |
| **3** | Відправити T5 Z5h,MK5h ,d,x |
| **4** | Отримати Rh,MOh,Ch,x від T5 |
| **5** | Uh = (MKh \* MO) |
| **6** | Ah = x \* Rh + d \* C \* Uh |
| **7** | Отримання A5h від T5 |
| **8** | Вивід А |

|  |  |
| --- | --- |
| **№** | **Задача *Т2*** |
| **1** | Отримати Mkh , Zh , d, x від T3 |
| **2** | x = min(min(Zh),x) |
| **3** | Відправити T3 x |
| **4** | Отримати Rh,MOh,Ch,x від T3 |
| **5** | Uh = (MKh \* MO) |
| **6** | Ah = x \* Rh + d \* C \* Uh |
| **7** | Відправити T3 Ah |

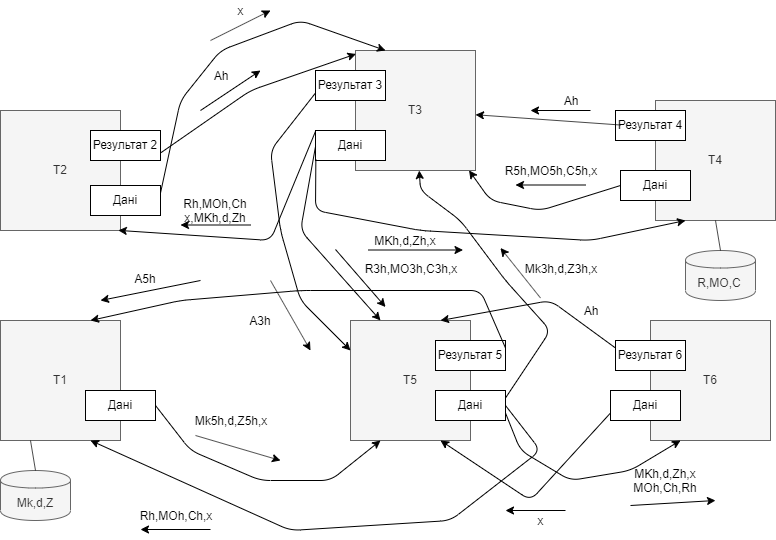
|  |  |
| --- | --- |
| **№** | **Задача *Т3*** |
| **1** | Отримати Mk5h,d,Z5h,x від T5 |
| **2** | x = min(Zh) |
| **3** | Відправити T2 Zh,MKh ,d,x |
| **4** | Отримати x від T2 |
| **5** | Відправити Mkh,d,Zh,x до T4 |
| **6** | Отримати від T4 R3h,MO3h,C3h,x |
| **7** | Відправити Rh,MOh,Ch ,x, T2 |
| **8** | Відправити R3h,MO3h,C3h ,x, T5 |
| **9** | Uh = (MKh \* MO) |
| **10** | Ah = x \* Rh + d \* C \* Uh |
| **11** | Отримати від T4 A3h |
| **12** | Отримати від T2 Ah |
| **13** | Відправити T5 A3h |

|  |  |
| --- | --- |
| **№** | **Задача *Т4*** |
| **1** | Введення R,MO,C |
| **2** | Отримання Mkh,d,Zh,z від T3 |
| **3** | x = min(Zh) |
| **4** | Відправити T3 R5h,C5h,MO5h,x |
| **5** | Uh = (MKh \* MO) |
| **6** | Ah = x \* Rh + d \* C \* Uh |
| **7** | Відправити Ah до А3 |

|  |  |
| --- | --- |
| **№** | **Задача *Т5*** |
| **1** | Отримати Mk5h,d,Z5h,x від T1 |
| **2** | x = min(Zh) |
| **3** | Відправити T6 Zh,MKh ,d,x |
| **4** | Отримати x від T6 |
| **5** | Відправити Mk3h,d,Z3h,x до T3 |
| **6** | Отримати від T3 R3h,MO3h,C3h,x |
| **7** | Відправити T3 R3h,MO3h,C3h ,x, T6 |
| **8** | Відправити T3 R3h,MO3h,C3h ,x, T1 |
| **9** | Uh = (MKh \* MO) |
| **10** | Ah = x \* Rh + d \* C \* Uh |
| **11** | Отримати від T3 A3h |
| **12** | Отримати від T6 Ah |
| **13** | Відправити T1 A5h |

|  |  |
| --- | --- |
| **№** | **Задача *Т6*** |
| **1** | Отримання Mkh, Zh ,d,x від T5 |
| **2** | x = min(min(Zh),x) |
| **3** | Відправити x доT5 |
| **4** | Отримати Rh,MOh,Ch,x |
| **5** | Uh = (MKh \* MO) |
| **6** | Ah = x \* Rh + d \* C \* Uh |
| **7** | Відправити T5 Ah |

**Етап 3. Розроблення структурної схеми взаємодії задач.**



**Умовні позначення:**

А – вектор розміру N

R – вектор розміру N

C – вектор розміру N

Z – вектор розміру N

МO – матриця (NxN)

МK – матриця (NxN)

x – мінімум вектора Z

*# lab5 python MPI*

*# Студента групи СП-424*

*# Єлістратова Кирила*

*# 01.12.19*

*from mpi4py import MPI*

*# Set The Comunicator*

*comm = MPI.COMM\_WORLD*

*# End Set The Comunicator*

*# Enter number of Task*

*rank = comm.Get\_rank()*

*# Enter number of Task*

*# T1 = 0 T2 = 1 T3 = 2 T4 = 3 T5 = 4 T6 = 5*

*#T1*

*if rank == 0:*

*# INIT DATA*

*P = 6*

*N = 36*

*h = N / P*

*MK = [1 in range(N \* N)]*

*d = 1*

*Z = [1 in range(N)]*

*Am = []*

*# END INIT DATA*

*# Begin Calculate min(Z)*

*x = min(Z[0:h])*

*# END Calculate min(Z)*

*# Message d,Z,MK to T5*

*data = [Z[h:N],d,MK[h \* N:N\*N],x]*

*# Message d,Z,MK to T5*

*comm.send(data, dest=5, tag=1)*

*#GET X*

*my\_full\_data = comm.recv(source=4,tag=)*

*for i in range(h \* 3,h\*4):*

*tmp = 0*

*for j in range(N):*

*MU[i][j] = MO[i][j] \* MK[j][i]*

*for k in range(N):*

*tmp += C[k] \* MU[i]*

*Am[i] = x \* R[h] + d \* tmp*

*tmp\_A = comm.recv(source=4,tag=16)*

*A = [i for i in tmp\_A,i for i in Am]*

*#OUTPUT*

*print(A)*

*#END OUTPUT*

*#END T1*

*#T5*

*elif rank == 4:*

*# INIT DATA*

*P = 6*

*N = 36*

*h = N / P*

*# END INIT DATA*

*# GET DATA FROM T1*

*my\_data = comm.recv(source=0, tag=1)*

*# END GET DATA FROM T1*

*Z = my\_data[0]*

*d = my\_data[1]*

*MK = my\_data[2]*

*tmp\_x = my\_data[3]*

*Am = []*

*# Begin Calculate min(Z)*

*x = min(Z[0:h])*

*x = min(x,tmp\_x)*

*# END Calculate min(Z)*

*data1 = [Z[h:h\*2],d,MK[h\*N :(h\*2)\*N],x]*

*# Send to t6*

*comm.send(data1,dest=5, tag=2)*

*# END Send to t6*

*# Recieve from t6*

*x = comm.recv(source=0, tag=11)*

*# END Recieve from T6*

*data2 = [Z[h\*2:N],d, MK[(h\*2) \* N: N\*N],x]*

*# Send to t3*

*comm.send(data2,dest=2, tag=3)*

*# END Send to t3*

*# RECV FOR X*

*last\_data = comm.recv(source=2,tag=8)*

*# END RECV FOR X*

*comm.send(last\_data,dest=5,tag=9)*

*comm.send(last\_data,dest=0,tag=10)*

*for i in range(h \* 3,h\*4):*

*tmp = 0*

*for j in range(N):*

*MU[i][j] = MO[i][j] \* MK[j][i]*

*for k in range(N):*

*tmp += C[k] \* MU[i]*

*Am[i] = x \* R[h] + d \* tmp*

*A2 = comm.recv(source=2,tag=15)*

*A3 = comm.recv(source=5,tag=14)*

*A\_last = [i for i in A2,i for i in Am,i for i in A3]*

*comm.send(A\_last,source=0,tag=16)*

*#END T5*

*# T3*

*elif rank == 2:*

*# INIT DATA*

*P = 6*

*N = 36*

*h = N / P*

*# END INIT DATA*

*# GET DATA FROM T1*

*my\_data = comm.recv(source=4, tag=)*

*# END GET DATA FROM T1*

*Z = my\_data[0]*

*d = my\_data[1]*

*MK = my\_data[2]*

*tmp\_x = my\_data[3]*

*Am = []*

*#Get Min X*

*x = min(tmp\_x,min(Z[0:h]))*

*#END Get Min X*

*data1 = [Z[h:h\*2],d,MK[h\*N :(h\*2)\*N],x]*

*# Send to t2*

*comm.send(data1,dest=1, tag=4)*

*# END Send to t2*

*x = comm.recv(source=1, tag=12)*

*data2 = [Z[h\*2:N],d, MK((h\*2) \* N,N\*N,my\_data[2]),x]*

*# Send to t4*

*comm.send(data2,dest=3, tag=5)*

*# END Send to t4*

*last\_data = comm.recv(source=3,tag=6)*

*R = last\_data[0]*

*C = last\_data[1]*

*MO = last\_data[2]*

*comm.send(last\_data,dest=4,tag=8)*

*comm.send(last\_data,dest=1,tag=7)*

*for i in range(h \* 2,h\*3):*

*tmp = 0*

*for j in range(N):*

*MU[i][j] = MO[i][j] \* MK[j][i]*

*for k in range(N):*

*tmp += C[k] \* MU[i]*

*Am[i] = x \* R[h] + d \* tmp*

*A2 = comm.recv(source=3,tag=12)*

*A3 = comm.recv(source=1,tag=13)*

*A\_last = [i for i in A2,i for i in Am,i for i in A3]*

*comm.send(A\_last,dest=4,tag=14)*

*#END T3*

*#T2*

*elif rank == 1:*

*# INIT DATA*

*P = 6*

*N = 36*

*h = N / P*

*# END INIT DATA*

*# GET DATA FROM T1*

*my\_data = comm.recv(source=4, tag=2)*

*# END GET DATA FROM T1*

*Z = my\_data[0]*

*d = my\_data[1]*

*MK = my\_data[2]*

*tmp\_x = my\_data[3]*

*# Calculate min(Z)*

*x = min(Z,tmp\_x)*

*# End Calculate min(Z)*

*comm.send(x,dest=2,tag=12)*

*last\_data = comm.recv(source=2,tag=7)*

*R = last\_data[0]*

*C = last\_data[1]*

*MO = last\_data[2]*

*for i in range(h,h2):*

*tmp = 0*

*for j in range(N):*

*MU[i][j] = MO[i][j] \* MK[j][i]*

*for k in range(N):*

*tmp += C[k] \* MU[i]*

*A[i] = x \* R[h] + d \* tmp*

*comm.send(A,dest=2,tag=13)*

*#END T2*

*#T4*

*elif rank == 3:*

*# INIT DATA*

*P = 6*

*N = 36*

*h = N / P*

*R = [1 in range(N)]*

*C =[1 in range(N)]*

*MO = [1 in range(N \* N)]*

*MU = [1 in range(N \* N)]*

*A = []*

*# END INIT DATA*

*#comm.send(data1,dest=1, tag=2)*

*# GET DATA FROM T3*

*my\_data = comm.recv(source=2, tag=5)*

*# END GET DATA FROM T3*

*Z = my\_data[0]*

*d = my\_data[1]*

*MK = my\_data[2]*

*tmp\_x = mydata[3]*

*# Calculate min(Z)*

*x = min(min(Z),tmp\_x)*

*# End Calculate min(Z)*

*# Begin Sending*

*data1 = [R[h:N],C[h:N],MO,x]*

*comm.send(data1,dest=2, tag=6)*

*#End Sending*

*for i in range(h):*

*tmp = 0*

*for j in range(N):*

*MU[i][j] = MO[i][j] \* MK[j][i]*

*for k in range(N):*

*tmp += C[k] \* MU[i]*

*A[i] = x \* R[h] + d \* tmp*

*comm.send(A,dest=2,tag=13)*

*#END T4*

*#T6*

*elif rank == 5:*

*# INIT DATA*

*P = 6*

*N = 36*

*h = N / P*

*MU = []*

*A = []*

*# END INIT DATA*

*# GET DATA FROM T5*

*my\_data = comm.recv(source=4, tag=3)*

*# END GET DATA FROM T3*

*Z = my\_data[0]*

*d = my\_data[1]*

*MK = my\_data[2]*

*tmp\_x = my\_data[3]*

*# Calculate min(Z)*

*x = min(my\_data[0])*

*x = min(x,tmp\_x)*

*# End Calculate min(Z)*

*comm.send(x,dest=4,tag=11)*

*last\_data = comm.recv(source=4,tag=9)*

*R = last\_data[0]*

*C = last\_data[1]*

*MO = last\_data[2]*

*for i in range(h\*4,h \* 5):*

*tmp = 0*

*for j in range(N):*

*MU[i][j] = MO[i][j] \* MK[j][i]*

*for k in range(N):*

*tmp += C[k] \* MU[i]*

*A[i] = x \* R[h] + d \* tmp*

*comm.send(A,dest=4,tag=15)*

*#END T6*