**Quiz CMPE-523 11.06.2020 (50 points, 110 min)**

St. Name, Surname\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ St.Id#\_\_\_\_\_\_\_\_\_\_\_\_\_

Instructor Alexander Chefranov

Open book, open notes

*Totally 5 tasks, 50 points, 2 pages*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Task 1 | Task 2 | Task 3 | Task 4 | Task 5 |
|  | 10 | 10 | 10 | 10 | 10 |
| Abdolrahman | 7 | 0 | 0 | 0 | 9 |
| Mustafa | 3 | 3 | 0 | 2 | 2 |
| Seyed Javad | 8 | 3 | 0 | 0 | 9 |
| Leila | 2 | 3 |  |  |  |

**Task 1. (10 points).** Consider the following arithmetic expression: $A∙B∙\left(\frac{C}{D}+A\right)-D∙E^{2}+B$. Draw a computation tree of the expression. Find the size and depth of the tree. Write out a sequence of machine-level instructions for its calculation. Draw a time diagram of execution of the instructions assuming the computer system has two adders/subtractors, and two multipliers/divisors. Assume the time for the operations performing is

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Operation | Addition | Subtraction | Multiplication | Division |
| Time | 1 | 2 | 3 | 4 |

Give necessary explanations.

B

B

D

B

B

D

C

A

A

Size=8, depth=5

1. T1=A\*B
2. T2=C/D
3. T3=T2+A
4. T3=T2\*T3
5. T4=B\*B
6. T5=D\*T4
7. T5=T3-T5
8. T5=T5+B

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| AS1 |  |  |  |  | S3 |  |  |  |  | S7 | S8 |
| AS2 |  |  |  |  |  |  |  |  |  |  |  |  |
| MD1 | S1 | S5 | S6 |  |  |  |
| MD2 | S2 |  | S4 |  |  |  |  |

**Task 2. (10 points).** Draw a computation graph, similar to that shown in Fig. 2-5 below, solving the parallel prefix problem by Upper-Lower Prefix Algorithm for 11 data items arranged as array, A[11], and 11 processing elements. Give necessary explanations.



The graph is built using the following decomposition: (((2+1)+(2+1))+((2+1)+(1+1)))

**Task 3. (10 points).** For the computational graph in Fig. 2-5, calculate speedup and efficiency for the number of processors, $p=5$. Give necessary explanations.

|  |  |  |  |
| --- | --- | --- | --- |
| P1 | V2’=V1+V2 | V3’=V2’+V3 | V5’=V4’+V5 |
| P2 | V4’=V3+V4 | V4’=V2’+V4’ | V6’=V4’+V6’ |
| P3 | V6’=V5+V6 | V7’=V6’+V6 | V7’=V4’+V7’ |
| P4 | V8’=V7+V8 | V8’=V6’+V8’ | V8’=V4’+V8’ |
| P5 |  |  |  |

Speedup(5)=T(1)/T(5)=7/3; Efficiency(5)=Speedup(5)/5=7/15. T(1)=7 for the sequential algorithm of solving the problem. It is not equal to the size, 12, of the graph in Fig. 2-5.

**Task 4. (10 points).** Using associativity, draw the flattest possible dependence graph for the following calculation

.$\sum\_{i=1}^{3}\sum\_{,j=1}^{3}(A\_{i}+A\_{j})/B\_{ij}$

Write SIMD pseudocode for its calculation. Assume that addition takes 1 time unit, multiplication takes 3 time units, and division takes 4 time units. What is the minimal number  of processors providing maximal performance for that program? Estimate speedup and efficiency for that number  of processors.

Denote by a node in the graph operation Op(a,b,c)=(a+b)/c. Then the graph is

`

A1

A2

A3

B11

B12

B13

b21

B22

B23

B31

B32

B33

$π=9$, Speedup=(9\*1+9\*4+8\*1)/(1+4+4\*1)=53/9; Efficiency=Speedup/9=53/81

**Task 5. (10 points).** Consider the code below

X[i]=c[i], (1<=i<=n);

For j:=1 step 1 until n-1

 X[i]:=x[i]+A[I,j]\*x[j], (j+1<=i<=min(j+m,n));

What problem is solved by the code? Linear recurrence problem

Assuming a SIMD computer with the distributed memory has N=5=n, m=3 processing elements, show memory allocation for the code. Trace the code. Assume C=(3,1,1,2,1), A=$\left|\begin{array}{c}\begin{matrix}\begin{matrix}0&0\end{matrix}&\begin{matrix}0&\begin{matrix}0&0\end{matrix}\end{matrix}\end{matrix}\\\begin{matrix}\begin{matrix}2&0\\3&2\end{matrix}&\begin{matrix}0&\begin{matrix}0&0\end{matrix}\\0&\begin{matrix}0&0\end{matrix}\end{matrix}\\\begin{matrix}1&1\\0&2\end{matrix}&\begin{matrix}3&\begin{matrix}0&0\end{matrix}\\1&\begin{matrix}4&0\end{matrix}\end{matrix}\end{matrix}\end{array}\right|$. Rewrite the code using y=broadcast(x) operation for broadcasting a scalar x to the local variable y of all the processing elements.

N=5 processing elements (PE)

Allocation of the data over PEs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Address | PE1 | PE2 | PE3 | PE4 | PE5 |
| 0 C | 3 | 1 | 1 | 2 | 1 |
| 1 X |  |  |  |  |  |
| 2 A(\*,1) | 0 | 2 | 3 | 1 | 0 |
| 3 A(\*,2) | 0 | 0 | 2 | 1 | 2 |
| 4 A(\*,3) | 0 | 0 | 0 | 3 | 1 |
| 5 A(\*,4) | 0 | 0 | 0 | 0 | 4 |
| 6 A(\*,5) | 0 | 0 | 0 | 0 | 0 |
| 7 t |  |  |  |  |  |

Tracing

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| j | t | X1 | X2 | X3 | X4 | X5 |
|  |  | 3 | 1 | 1 | 2 | 1 |
| 1 | 3 |  | 1+2\*3=7 | 1+3\*3=10 | 2+1\*3=5 |  |
| 2 | 7 |  |  | 10+2\*7=24 | 5+1\*7=12 | 1+2\*7=15 |
| 3 | 24 |  |  |  | 12+3\*24=84 | 15+1\*24=39 |
| 4 | 84 |  |  |  |  | 39+4\*84=375 |

Check the calculations:

X1=c1=3; x2=c2+a21\*x1=1+2\*3=7; x3=c3+a31\*x1+a32\*x2=1+3\*3+2\*7=24; x4=c4+a41\*x1+a42\*x2+a43\*x3=2+1\*3+1\*7+3\*24=84; x5=c5+a52\*x2+a53\*x3+a54\*x4=1+2\*7+1\*24+4\*84=375

X[i]=c[i], (1<=i<=n);

For j:=1 step 1 until n-1 begin

 Y=broadcast(x[j]);

 X[i]:=x[i]+A[I,j]\*Y, (j+1<=i<=min(j+m,n));

End;