Concurrent Programming Coursework

Summary: You are required to implement the parallel *prefix scan* algorithm in c/mpi:

Specification:

- To have keyboard input of the number, n, of elements in the sequence
- Process 0 will generate n random integers and broadcast these to the other processes as appropriate
- Process 0 prints the input and output sequences onto the screen.

Note: use of MPI_Scan is explicitly forbidden.

Methodology:

There will be an in-class discussion of the algorithm in which we shall work together to identify the key issues and details of the algorithm – be prepared to ask questions in that session.

Deliverables:

Assignment Project Exam Help

1) A single source file prefix.c that can be compiled under MSYS2 using mpicc. This source file will contain all necessary c/mpi code. Codes provided for other frameworks (e.g. implemented using Microsoft Visual Studio with MPI are NOT ACCEPTABLE and will be awarded a mark of 0)

https://powcoder.com

Your code must be properly documented to permit an experienced coder, who is unfamiliar with the prefix scan algorithm to understand and be able to modify or upgrade it.

Add WeChat powcoder

- Good quality in-code commentary is required

- Explicit testing showing sets of inputs and expected outputs is required

2) A very brief pdf report highlighting any significant code features to which you wish to draw attention and the results of your testing to convince the client of correct functionality and any comments on the limitations and scope for upgrade of the code.

Submission is only accepted via the assessment link on the Moodle page.

Marks and feedback will be provided within 15 working days of the submission deadline and each student will receive their completed copy of the marking rubric posted on Moodle. Further feedback can subsequently be requested by email.

Plagiarism: Students will naturally discuss the assignment between themselves, however each student's submission must be wholly their own work. It is easier than you might think to spot both students sharing code and the use of third-party code downloaded from the internet etc. The full force of the University's academic misconduct regulations will be invoked for any infringements.

The Prefix Scan Algorithm

Definition: Given a sequence, ***a***, of input values $(a_0, a_1, a_2, a_3, a_4 \ldots a_{N-1})$

the output sequence, ***b***, is defined as $(b_0, b_1, b_2, b_3, b_4 \ldots b_{N-1})$ where $b_i = \sum_{j=0}^{i} a_j$

We note that $b_i = b_{i-1} + a_i$

Example: ***a***$= (1, 2, 3, 4, 5, 6, 7, 8)$ then ***b***$= (1, 3, 6, 10, 15, 21, 28, 36)$

A simple serial C implementation of this algorithm might be

```
void prefix_scan(int n, int a[], int b[]){
        b[0]=a[0];
        for(int i=1;i<n;++i) b[i]=b[i-1]+a[i];
}
```

Parallel Prefix Scan Algorithm

Initialise b with ***a***

Assignment Project Exam Help

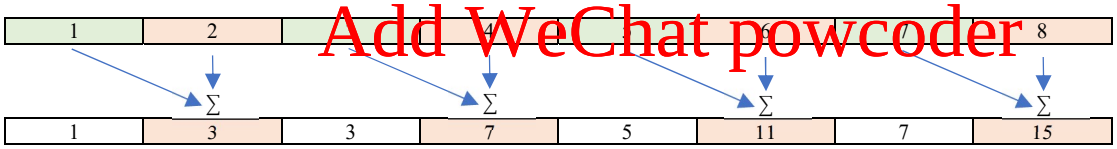
Up Phase

Note those elements in cells of index 2, 4, 6, 8 …,i.e. spaced by steps of 2

https://powcoder.com

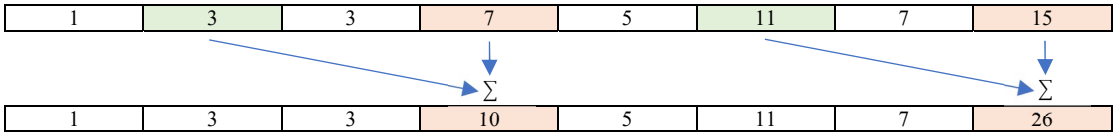
Each such cell's value is replaced by the sum of the current value and the value in the cell 1 place before it. All other cells keep the same value

Add WeChat powcoder



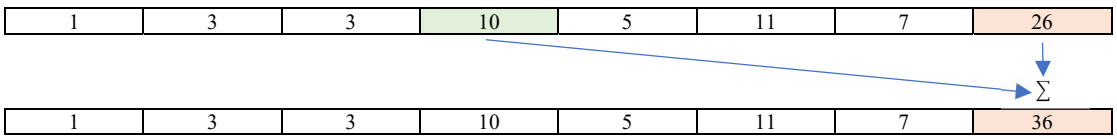
Note those elements in cells of index 4, 8, 12 ,16 …,i.e. spaced by steps of 4

Each such cell's value is replaced by the sum of the current value and the value in the cell 2 places before it. All other cells keep the same value



Note those elements in cells of index 4, 8, 12 ,16 …,i.e. spaced by steps of 8

Each such cell's value is replaced by the sum of the current value and the value in the cell 4 places before it. All other cells keep the same value

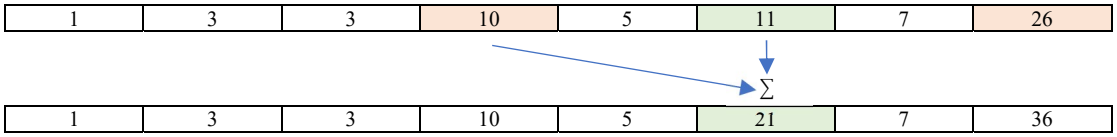


This is continued until, as just reached in this example, there are no further changes.

Down Phase

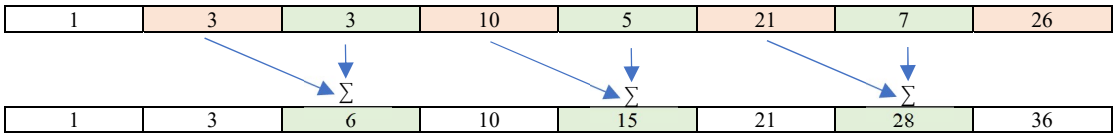
Note those elements in cells of index 4, 8, 12 ,16 … i.e. spaced by steps of 4 (i.e. one less than the highest reached in the up phase)

Each such cell's value contributes to a sum of the current value and the value in the cell 2 places after it. All other cells keep the same value



Note those elements in cells of index 2, 4, 6 ,8 …,i.e. spaced by steps of 2

Each such cell's value is contributes to a sum of the current value and the value in the cell 1 places after it. All other cells keep the same value



We could compress this into the following to succinctly show the evolution to the final result.

Assignment Project Exam Help

1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	Noting cells spaced by steps of 2
1	3	3	7	5	11	7	15	Noting cells spaced by steps of 4
1	3	3	10	5	11	7	26	Noting cells spaced by steps of 8
1	3	3	10	5	11	7	36	

https://powcoder.com

1	3	3	10	5	11	7	36	Noting cells spaced by steps of 8
1	3	3	10	5	11	7	36	Noting cells spaced by steps of 4
1	3	3	10	5	21	7	36	Noting cells spaced by steps of 2
1	**3**	**6**	**10**	**15**	**21**	**28**	**36**	

Add WeChat powcoder

Outlined dells are the noted ones per round and shaded cells are those whose value has changed

A different example illustrates the need to *pad with zeros* the total number of elements to a power of 2

3	**7**	**1**	**11**	**8**	**20**	**4**	**-1**	**4**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
3	10	1	12	8	28	4	3	4	4	0	0	0	0	0	0
3	10	1	22	8	28	4	31	4	4	0	4	0	0	0	0
3	10	1	22	8	28	4	53	4	4	0	4	0	0	0	4
3	10	1	22	8	28	4	53	4	4	0	4	0	0	0	57

3	10	1	22	8	28	4	53	4	4	0	4	0	0	0	57
3	10	1	22	8	28	4	53	4	4	0	4	0	0	0	57
3	10	1	22	8	28	4	53	4	4	0	57	0	0	0	57
3	10	1	22	8	50	4	53	4	57	0	57	0	57	0	57
3	**10**	**11**	**22**	**30**	**50**	**54**	**53**	**57**	**57**	**57**	**57**	**57**	**57**	**57**	**57**