



МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М. В. ЛОМОНОСОВА

Факультет вычислительной математики и кибернетики

Отчет по теоретическому заданию

«Суперкомпьютерное моделирование и технологии»

ВАРИАНТ 77

Студент 615 группы

Егоров Кирилл

Москва, 2022

Содержание

1	Постановка задачи
---	-------------------

3

1 Постановка задачи

В качестве условия задачи выступает следующий фрагмент кода программы на языке C:

```
for(i = 2; i <= n+1; ++i)
    C[i] = C[i - 2] + D[i];
for(i = 2; i <= n+1; ++i)
    for(j = 2; j <= m+1; ++j)
        B[i][j] = B[i][j - 1] + C[n + 1];
for(i = 2; i <= n+1; ++i){
    A[i][1][1] = C[i];
    for(j = 2; j <= m+1; ++j){
        for(k = 1; k <= n; ++k)
            A[i][j][k] = A[i][j - 1][k - 1] + A[i][j][k];
    }
}
```

В рамках задания необходимо исследовать информационную структуру указанного фрагмента, то есть выявить имеющиеся в ней зависимости по данным и их характер, после чего составить описание информационной структуры на языке разметки Algolang. Необходимо привести значения следующих величин для данного алгоритма (в зависимости от параметров программы n и m):

1. Число вершин в информационном графе фрагмента (последовательная сложность);
2. Длина (число дуг) критического пути в информационном графе (параллельная сложность);
3. Ширина (максимальное число вершин на ярусе) ярусно-параллельной формы (в тексте дайте пояснения, для какой именно ЯПФ приведено значение ширины);
4. Максимальная глубина вложенности циклов;

5. Число различных типов дуг (тип дуг определяется направляющим вектором и длиной при фиксированных значениях параметров);
6. Наличие длинных дуг (т.е. дуг, длина которых зависит от внешних параметров).

После исследования информационной структуры требуется разметить параллельные циклы заданного фрагмента программы с использованием директивы OpenMP `#pragma omp parallel for`.

2 Построение информационного графа

В рамках задачи был построен информационный граф для значений параметров $n = 4$, $m = 3$. Для этого было написано следующее описание графа на языке Algolang:

```
<?xml version="1.0"?>
<algo>
  <params>
    <param name="n" type="int" value="4"></param>
    <param name="m" type="int" value="3"></param>
  </params>
  <block id="0" dims="1">
    <arg name="i" val="2..n+1"></arg>
    <vertex condition="" type="1">
      <in src="i-2"></in>
    </vertex>
  </block>
  <block id="1" dims="2">
    <arg name="i" val="2..n+1"></arg>
    <arg name="j" val="2..m+1"></arg>
    <vertex condition="" type="1">
      <in src="i, j-1"></in>
      <in bsrc="0" src="n+1"></in>
    </vertex>
```

```

</block>
<block id="2" dims="3">
  <arg name="i" val="2..n+1"></arg>
  <arg name="j" val="1..m+1"></arg>
  <arg name="k" val="1..n"></arg>
  <vertex condition="(j==1) and (k==1)" type="1">
    <in bsrc="0" src="i"></in>
  </vertex>
  <vertex condition="(j>1)" type="1">
    <in src="i, j-1, k-1"></in>
    <in src="i, j, k"></in>
  </vertex>
</block>
</algo>

```

Данное описание было построено в системе Algoload. Ниже представлена визуализация в проекциях на плоскости XY , YZ , XZ , а также на плоскость, с которой субъективно лучше всего видно структуру информационного графа рассматриваемого алгоритма.