

**Отчет**

**Параллельная программа на OpenMP, которая реализует  
однокубитное квантовое преобразование.**

**Факультет:** Вычислительной математики и кибернетики

**Кафедра:** Суперкомпьютеров и квантовой информатики

**Группа:** 323

**Студент:** Тыркалов Евгений Олегович

### Задача:

1. Реализовать параллельную программу на C++ с использованием OpenMP, которая выполняет однокубитное квантовое преобразование над вектором состояний длины  $2^n$ , где  $n$  – количество кубитов, по указанному номеру кубита  $k$ .
2. Определить максимальное количество кубитов, для которых возможна работа программы на системе Polus.
3. Протестировать программу на системе Polus, используя преобразование Адамара по номеру кубита.

### Описание алгоритма:

Математическая постановка:

Имеется комплексный входной вектор (массив) размерности  $2^n$ :  $\{a_i\} = \{a_0, a_1, \dots, a_{2^n-1}\}$ ;  $n$  – параметр задачи (число кубитов). Над такими векторами нам необходимо производить так называемые однокубитные операции. Обе эти операции переводят вектор в новый вектор такой же размерности (длины массива). Однокубитная операция задается двумя параметрами: комплексной матрицей размера  $2 \times 2$  и числом от 1 до  $n$  (данный параметр обозначает номер кубита, по которому проводится операция). Итак, дана комплексная матрица:

$$U = \begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix}$$

и  $k$  – номер индекса от 1 до  $n$  (номер кубита).

Такая операция преобразует вектор  $\{a_{i_1, i_2, \dots, i_n}\}$  в  $\{b_{i_1, i_2, \dots, i_n}\}$ , где все  $2^n$  элементов нового вектора вычисляются по следующей формуле:

$$b_{i_1, i_2, \dots, i_k, \dots, i_n} = \sum_{j_k=0}^1 u_{j_k, i_k} a_{i_1, i_2, \dots, i_k, \dots, i_n}$$

Преобразование Адамара задается следующей матрицей:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

## Результаты выполнения:

количество кубитов	количество процессоров	время работы программы (сек)			Ускорение		
		1 кубит	10 кубит	20 кубит	1 кубит	10 кубит	последний кубит
20	1	0,096259	0,096132	0,09478	1	1	1
	2	0,052591	0,051892	0,051336	1,830332	1,852539	1,846267
	4	0,032511	0,037334	0,028808	2,960813	2,574918	3,290058
	8	0,017824	0,018232	0,018229	5,400527	5,272707	5,199407
	32	0,011589	0,011854	0,011852	8,305962	8,109376	7,996641
	64	0,009689	0,009910	0,009909	9,934638	9,699504	9,564664
	128	0,008658	0,008856	0,008855	11,11757	10,85444	10,703546
	160	0,008328	0,008299	0,008275	11,55752	11,58279	11,452516
24	1	1,5372	1,53917	1,52544	1	1	1
	2	0,81086	0,81086	0,816698	1,895764	1,898194	1,867814
	4	0,469848	0,475916	0,460665	3,271696	3,234121	3,311386
	8	0,292084	0,315461	0,286755	5,262869	4,879113	5,319663
	32	0,194305	0,209856	0,190760	7,911268	7,334397	7,996641
	64	0,162450	0,175452	0,15948	9,462550	8,772563	9,564664
	128	0,145165	0,156784	0,14251	10,58927	9,817128	10,70354
	160	0,136483	0,1436	0,13018	11,26292	10,71816	11,71739
28	1	24,2861	24,2628	24,3464	1	1	1
	2	13,0918	12,9786	12,9719	1,855061	1,869446	1,876856
	4	7,42278	7,33705	7,33932	3,271833	3,306887	3,317255
	8	4,54453	4,51497	4,5097	5,344028	5,373856	5,398673
	32	3,06809	3,04813	3,04457	7,915700	7,959881	7,996641
	64	2,56512	2,54882	2,54545	9,467851	9,520695	9,564664
	128	2,29217	2,2717	2,27461	10,59520	10,65434	10,70354
	160	2,12354	2,0553	2,04739	11,43661	11,80499	11,89143
30	1	103,49	97,567	97,6294	1	1	1
	2	55,3426	52,5547	52,4574	1,869988	1,856484	1,861117
	4	30,9996	29,4618	29,6792	3,338430	3,311644	3,289488
	8	20,346	18,5931	18,0467	5,086503	5,247484	5,409820
	32	13,7643	12,5784	12,2088	7,518724	7,756681	7,996641
	64	11,5077	10,5163	10,2073	8,993034	9,277651	9,564664
	128	10,2833	9,3973	9,1212	10,06385	10,382358	10,70354
	160	9,1212	8,8519	8,8817	11,34604	11,022140	10,99219

График ускорения для 30 кубитов при изменении 10го

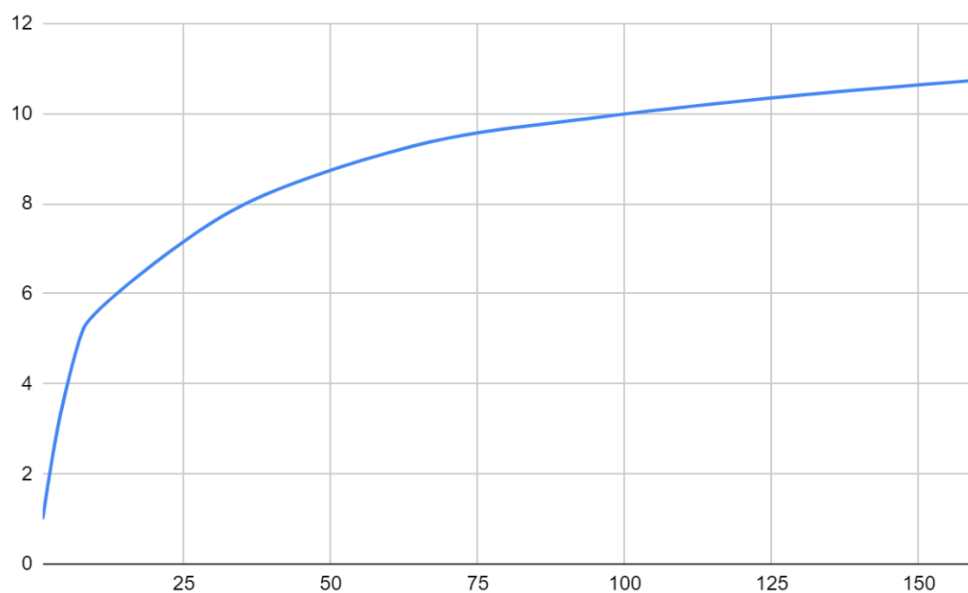


График иллюстрирует экспоненциальную зависимость ускорения от числа потоков. Наличие “Плато” обусловлено ограниченностью эффективности распараллеливания.