Математичка гимназија

Матурски рад
из предмета физика

Квантни алгоритми

Ментор
др Игор Салом

Ученик
Лазар Галић, IV$_{Д}$

Београд, јун 2019.

Садржај

Напомена:

Наслови у овом раду означени су на следећи начин:

- Поглавља су означена бројевима - **1**, **2**...
- Додаци су означени великим азбучним словима - **А**, **Б**...
- Садржај и литература немају додатних ознака

1 Увод

Рачунар је настао да решава проблеме који раније нису постојали

Бил Гејтс

Квантни рачунари су релативно нов појам у науци. Иако се за њих зна већ око пола века, ово поље науке је и даље у повоју развоја. Иако је већ познат не мали број начина примене квантних рачунара, они су још увек углавном хипотетски, и на праву, реалну примену квантних рачунара се још чека.

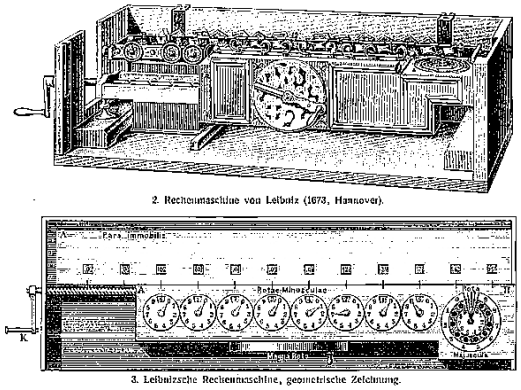
1.1 Кратка историја рачунарства

1645. Блез Паскал[1] осмишљава, а следеће године и конструише први механички калкулатор који може да сабира и одузима - *Паскалину*. Око 30 година касније, немачки научник Готфрид Лајбниц[2] осмислио је *степенасти цилиндар* (нем. *Staffelwalze*). То је била прва машина способна за извршавање све четири основне аритметичке операције. Иако технологија тог времена није имала довољну прецизност за овакву машину, конструисана су два модела, од којих је један сачуван до данас.

После скоро два века, 1820. године, Томас де Калмар[3] конструише први калкулатор довољне јачине и поузданости за комерцијалну употребу, *аритмометар*. Он је користио покретни акумулатор за рачунање производа и количника два дата броја. Две године касније, Чарлс Бебиџ[4] долази на идеју прављења *диференцијалне машине* - машине која може да процени вредности већине често коришћених функција, укључујући и логаритамске и тригонометријске функције. Бебиџ је направио само једну седмину првог прототипа *Диференцијалне машине бр. 1*, пре него што је британска влада, због велике цене пројекта узроковане ценом прецизних делова потребних за машину, отказала пројекат. 1991. године је, у част 200-те годишњице Бебиџовог рођења, на Универзитету у Сиднеју је направљена *Диференцијална машина бр. 2* према Бебиџевим нацртима, која и данас ради, и којом је показана исправност његових нацрта.

Током развоја *диференцијалне машине*, Бебиџ долази на идеју додавања још једне кључне идеје својој машини - програмабилност. 1834. године, почео је са радом на развоју ове идеје, и током наредне две године је описао готово све основне карактеристике модерног рачунара. Резултат свог рада назвао је *аналитичка машина*. Кључна идеја је била имплементација *бушених картица*, до тада коришћених за програмирање машина за ткање у текстилној индустрији, као складишта за програме. Ада Лавлејс[5] је 1843. године, током превођења чланка о *аналитичкој машини* с француског, записала алгоритам за рачунање Бернулијевих бројева. Сматра се да је ово први написани компјутерски програм, а да је самим тим и Ада прва програмерка.

[1] **Blaise Pascal** ***(1623-1662)*** - француски математичар
[2] **Gottfried Wilhelm Leibniz** ***(1646-1716)*** - немачки научник
[3] **Charles Xavier Thomas de Calmar** ***(1785-1870)*** - француски проналазач и предузетник
[4] **Charles Babbage** ***(1791-1871)*** - британски математичар и проналазач
[5] **Ada Byron Lovelace** ***(1815-1852)*** - британска математичарка и писац



Слика 1: *Лајбницов цилиндар*

Током Другог светског рата, долази до прве масовне употребе рачунара, пре свега за разбијање непријатељских шифара, али и за комплексне рачунске проблеме. 1941. године, у Берлину, Конрад Зусе[6] прави први дигитални, програмабилни, потпуно аутоматски рачунар *Z3*. Овај рачунар је користио бинарни систем, чувао је речи дужине 22 бинарне цифре, и радио до 5 операција у секунди. 1991. је доказана Тјуринг-комплетност овог рачунара, али само кроз рачунање свих могућих исхода, јер није било условног гранања. Срећом, овај пројекат није био сматран веома важним, и није нашао значајнију практичну употребу.

Паралелно овом пројекту, у чувеној британској тајној војној лабораторији за дешифровање нацистичких шифрованих порука *Блечли Парк* (*Bletchley Park*) се 1943. развија *Колос* (*Colossus*), први електронски дигитални програмабилни рачунар, који је заиста био Тјуринг комплетан (имао је могућност условног гранања). *Колос* је користио електронске цеви за логичке и нумеричке операције, а био је програмиран прекидачима. Овај рачунар је развијен за разбијање пресретнутих порука шифрираних непознатим уређајем[7]. 1946. године, Алан Тјуринг[8], један од главних чланова тима у Блечли Парку, даје први детаљан опис рачунара са програмима сачуваним у електронској меморији. Такође, он даје теоријску дефиницију и основу рачунара, као *Тјурингове комплетне машине* (касније ће бити више речи о овоме). Даљим развојем његове идеје, Џон фон Нојман[9] развија тзв. *Принстон архитектуру*, по којој се рачунари деле на:

1. *Процесорску јединицу*, која садржи тзв. *аритметичко-логичку јединицу* (за рачунање) и *процесорске регистре* (за складиштење параметара и резултата *аритметичко-логичке јединице*)
2. *Контролну јединицу*, која садржи тзв. *регистар инструкције* (за складиштење информација о инструкцији) и *бројач програма* (за информација о меморијској локацији следеће инструкције)
3. *Меморију* за чување података и инструкција
4. Спољно *складиште података*
5. Механизме за *улаз* и *излаз*

[6]**Konrad Zuse** ***(1910-1995)*** - немачки проналазач и инжењер
[7]Тјуринг је радио и на развоју *Бомбе*, уређаја за дешифровање порука са *Енигме*, која није програмабилан уређај.
[8]**Alan Turing** ***(1912-1954)*** - британски математичар
[9]**John von Neumann** ***(1903-1957)*** - мађарско-амерички математичар

И данас, више од 70 година касније, рачунари функционишу по истом принципу. Током овог времена, дошло је до незамисливог повећања моћи рачунара. Брзина рачунара расла је експоненцијално, заједно са меморијом. Хардверске компоненте постајале су све мање, са променом технологија од електронских цеви, преко транзистора, интегрисаних кола, све до данашњих процесора.

Данашњи рачунари имају невероватне могућности. Од симулација и компјутерских игрица са детаљима на ивици реалности, преко веома брзог рачунања изузетно комплексних решења математичких проблема, до никад боље повезаности уређаја, где се сваки уређај може повезати са сваким другим у секунди, било где у свету. Ипак, у последње време овај експоненцијални раст успорава, јер смо достигли толико мале димензије транзистора, да постулати класичне физике престају да важе. Можемо ли даље?

1.2 Кратка историја квантне физике

Још од првих цивилизација и открића писма, човечанство је тежило да објасни природне феномене. Велики број феномена били су посматрани и проучавани вековима, увек у покушају да се нађе што боље објашњење за те појаве. Човечанство је увек желело бољи одговор, и као решење настала је наука физика.

До краја XIX века, објашњена је већина природних феномена - гравитација, електрицитет, светлост, као и многи други. Чувеном лорду Келвину[10] се приписује изјава[11]: *...изгледа вероватно да је већина основних принципа универзума већ утемељена. Будућност физике крије се иза шесте децимале...* Као што се касније испоставило, није могао бити мање у праву.

Лудвиг Болцман[12] је 1877. претпоставио да је могуће да су енергетски нивои физичког система, као што је молекул, дискретни, односно да постоје одређене могуће вредности енергија, али да су оне квантоване - да су целобројно пута веће од дате константе. Неколико година касније, 1900. године, немачки физичар Макс Планк[13] извео је тзв. *Планков закон*, у ком је применио Болцманову дистрибуцију, која је последица квантоване природе енергије.

Током 1905. године, Алберт Ајнштајн[14] је објаснио *фотоелектрични ефекат* тако што је претпоставио да се светлост, али и свако друго електромагнетно зрачење, може поделити на коначан број *кванта енергије* који су локализоване тачке у простору, које се крећу у целини, без дељења, и могу се емитовати и апсорбовати искључиво у целини.

Ово је вероватно најважније откриће у физици XX века. Поменути *кванти енергије* касније су постали познати по имену *фотони*. Њихов проналазак је решио проблеме који су настајали као последица ранијих теорија, заснованих на континуалној природи енергије и електромагнетних таласа. Већ 1913. године, Нилс Бор[15] дао је објашњење за спектралне линије атома водоника, поново засновано на квантној природи енергије. Ипак, свака анализа квантне природе енергије до тада била је искључиво објашњење за неки феномен који се није могао објаснити класичним постулатима физике. Ово се назива и *старом квантном теоријом*.

[10] **William Thompson Kelvin** ***(1824-1907)*** - британски физичар

[11] Нешто слично изјавио је, у ствари, Алберт Мајкелсон (**Albert Michelson** ***(1852-1931)*** - амерички физичар) 1894. године, док се сматра да Келвин није рекао ништа слично.

[12] **Ludwig Boltzmann** ***(1844-1906)*** - аустријски физичар

[13] **Max Planck** ***(1858-1947)*** - немачки физичар

[14] **Albert Einstein** ***(1879-1955)*** - немачки теоријски физичар

[15] **Niels Bohr** ***(1885-1962)*** - дански физичар

1923. године, француски физичар Луј де Брољ[16] даје *Де Брољеву хипотезу*, која тврди да честице могу показивати својства таласа, али и таласи својства честица, као објашњења чувеног *Јунговог експеримента* (експеримента са два прореза - интерференције електромагнетних таласа). На овоме су Вернер Хајзенберг и Макс Борн засновали своју *дискретну* квантну механику са *Хајзенберговим принципом неодређености*, док је Ервин Шредингер постулирао *таласну* квантну механику, заједно са нерелативистичком *Шредингеровом једначином*. Шредингер је касније такође показао да су ови приступи еквивалентни.

Даљи развој физике заснивао се на откривању физике елементарних честица, као и објашњавању нових парадокса и феномена. Следећи велики искорак за квантно рачунарство било је откриће Ричарда Фејнмана[17] 1981. године, који је показао да је немогуће на ефикасан начин симулирати квантне системе на класичним рачунарима, и предложио је основни модел за квантни рачунар. 1994. Питер Шор[18] проналази алгоритам за брзу факторизацију природних бројева, чиме даје један од првих значајних начина примене квантних рачунара.1998. Лов Гровер[19] проналази алгоритам за брзу претрагу базе података, чиме додатно доприноси надмоћи квантних рачунара. Ипак, тренутно најмоћнији квантни рачунар има једва 70 кубита, највећи број факторизован квантним рачунаром је 143, а цена квантног рачунара је више десетина милиона долара. Дугачак пут је пред нама.

Како функционишу ови алгоритми, и због чега су толико бољи од класичних алгоритама? Како у ствари раде квантни рачунари? Шта је *бит*, а шта *кубит*? Како изгледа квантна логика? Одговоре на ова, али и многа друга питања, размотрићемо у овом раду. Почев од основних појмова квантне физике, доћи ћемо до начина за разбијање већине криптографских алгоритама и прављење екстремно брзих база података.

[16] **Louis de Broglie** ***(1892-1987)*** - француски физичар
[17] **Richard Feynman** ***(1918-1988)*** - амерички физичар
[18] **Peter Shor** ***(1959-)*** - амерички математичар
[19] **Lov Kuman Grover** ***(1961-)*** - индијско-амерички физичар

2 Основи квантног рачунарства

Мислим да са приличном сигурношћу могу да кажем да нико не разуме квантну механику.

Ричард Фејнман

На почетку, неопходно је увести неколико кључних појмова из математике како бисмо лакше анализирали појаве у квантној физици. Као што је већ поменуто у уводном делу, закони квантне физике не понашају се у складу са досадашњим посматрањем света као тродимензионалног Еуклидског простора, већ прате правила неких других простора, са другим бројем димензија. Ипак, утврђени закони класичне физике морају остати добре апроксимације за макроскопске феномене. Као решење овог проблема, уводимо следеће појмове:

2.1 Хилбертов простор. Диракова нотација

P је ***линеаран векторски простор*** уколико се састоји од скупа вектора $V_P = \{\phi, \psi, \chi, \ldots\}$ и скупа скалара $S_P = \{a, b, c, \ldots\}$, и ако за њега важе:

- *Правило сабирања два вектора*
- *Правило множења вектора скаларом*

Најпознатији пример векторског простора је *тродимензионални* ***Еуклидски простор*** E^3, у ком је скуп вектора представљен скупом свих линеарних комбинација ортонормалних вектора e_x, e_y, e_z, а скуп скалара је скуп реалних бројева $\mathbb{R}$. У општем случају, линеарни векторски простори се могу посматрати као генерализације управо овог простора у ком смо мислили да живимо, а који је за свакодневне прилике одлична апроксимација.

Генерализација Еуклидског простора на бесконачно много димензија, са скупом комплексних бројева $\mathbb{C}$ као скупом скалара, даје ***Хилбертов***[20] ***простор*** H, на ком се посматрају појаве у квантној физици. Као што ће се показати, квантне појаве се врло добро објашњавају проширењем на $\mathbb{C}$, као и проширењем на неограничени број димензија.

За Хилбертов простор H важи:

1. H је линеаран векторски простор
2. дефинисан је скаларни производ вектора (a, b);
3. H је комплетан - за сваки конвергентан низ у H, вредност којој конвергира такође припада H
4. дефинисано је растојање $d(a, b) = \sqrt{(a - b, a - b)}$

[20] **David Hilbert** ***(1862-1943)*** - немачки математичар

Ради лакше анализе Хилбертових простора, уводимо и појмове:

- **Оператор** $A : H \to H$ је математичко правило, које примењено на вектор ψ даје вектор $\psi' = A(\psi) \in H$.
- **Дуални Хилбертов простор** H_d, у коме је скуп вектора скуп свих линеарних трансформација $\phi : H \to \mathbb{C}$
- **Сопствени вектор** ψ и **сопствена вредност** $\lambda \in \mathbb{C}$ оператора A представљају једно од решења једначине:
$$A\psi = \lambda\psi$$
- **Дуалан векторски простор** V_d је векторски простор свих линеарних трансформација ϕ векторског простора V. У простору V_d важе сва својства векторског простора. Такође, ако је V Хилбертов простор, онда је и V_d такође Хилбертов простор.

1939. године, Пол Дирак[21] уводи нотацију:

- $|\psi\rangle \in H$ је **кет**, вектор посматраног Хилбертовог простора. Може се посматрати као ***колона-вектор***.
- $\langle\phi| \in H_d$ је **бра**, функција дуалног Хилб. простора која одговара вектору ϕ. Посматра се као ***врста-вектор***.
- $\langle\psi|\phi\rangle \in \mathbb{R}$ је **бракет**, скаларни производ вектора ψ и ϕ.
- $A : H \to H$ (аналогно се посматра и $A : H_d \to H_d$) је оператор над Хилбертовим простором, који се може представити као матрица.

Као што смо приметили, до сада смо посматрали да H и H_d имају дискретан скуп базних вектора, чији су остали вектори линеарне комбинације. Могуће је и посматрати Хилбертове просторе који имају континуалне базе, али нам за анализу рада квантних рачунара са аспекта логичких кола нису неопходни. Сходно томе, даље сматрамо да простори H и H_d имају дискретне базе, те се њихови *бра*-ови и *кет*-ови могу посматрати преко координата као на слици 2.

$$|\psi\rangle = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \end{pmatrix} \qquad \langle\phi| = \begin{pmatrix} \overline{b_1} & \overline{b_2} & \dots \end{pmatrix}$$

Кет - ***колона-вектор*** **Бра -** ***врста-вектор***

$$\langle\phi|\psi\rangle = \begin{pmatrix} \overline{b_1} & \overline{b_2} & \dots \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ \vdots \end{pmatrix} = \sum_{i=0}^{\infty} a_i \overline{b_i} \qquad A = \begin{pmatrix} A_{11} & A_{12} & \dots \\ A_{21} & A_{22} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

Бракет - ***скаларни производ*** **Оператор -** ***матрица***

Слика 2: *Приказ бра-а, кет-а, бракет-а и оператора - матрице*

[21] **Paul Dirac** ***(1902-1984)*** - енглески физичар

2.2 Оператори

Као што је већ поменуто, оператор O је математичко правило које, примењено на кет $|\psi\rangle$, односно бра $\langle\phi|$, даје нови кет $|\psi'\rangle$, односно бра $\langle\phi'|$, у складу са задатим математичким правилом.

За операторе важе одређена, лако доказива својства:

- Оператори не морају бити комутативни: $AB \not\equiv BA$
- Оператори су асоцијативни: $(AB)C = A(BC)$
- Оператори делују *с десна на лево* на кетове и *с лева на десно* на браове:

$$AB\,|\psi\rangle = A(B\,|\psi\rangle) \qquad\qquad \langle\phi|\,AB = (\langle\phi|\,A)B$$

- Када су између браа и кета, као коначан резултат се добија комплексан број[22]:

$$\langle\phi|\,A\,|\psi\rangle \in \mathbb{C}$$

- $|\psi\rangle\,\langle\phi|$ је линеарни оператор.
- За сваки кет постоји тачно један одговарајући бра:

$$|\psi\rangle \longleftrightarrow \langle\psi|$$

  Притом, њихове одговарајуће координате су једна другој комплексни конјугат:

$$\begin{pmatrix} a_1 \\ a_2 \\ \vdots \end{pmatrix} = \begin{pmatrix} \overline{a_1} & \overline{a_2} & \dots \end{pmatrix}$$

Сада, желимо да уопштимо ову корелацију између одговарајућих бра-кет парова. Посматрајмо шта се дешава уколико посматрамо транспонован оператор (оператор је матрица, па га можемо транспоновати) са комплексно коњугованим координатама: $A^\dagger = \overline{(A)^T}$, што називамо ***хермитска конјугација***. Међутим, ово није сасвим коректна дефиниција, па хермитску конјугацију дефинишемо на следећи начин:

- хермитска конјугација комплексног броја је његов комплексни конјугат: $a^\dagger = \overline{a}$
- хермитска конјугација кета (односно браа) је његов одговарајући бра (односно кет): $|\psi\rangle^\dagger = \langle\psi|$, $\langle\phi|^\dagger = |\phi\rangle$
- хермитска конјугација $A^\dagger$ оператора A је онај оператор, за који за свака два вектора $|\phi\rangle$, $|\psi\rangle$ важи:

$$\langle\psi|\,A^\dagger\,|\phi\rangle = \big(\langle\phi|\,A\,|\psi\rangle\big)^\dagger$$

[22]Редослед није битан: $\langle\phi|\,(A\,|\psi\rangle) = (\langle\phi|\,A)\,|\psi\rangle$

Постоје оператори чије су хермитске конјугације једнаке њима самима: $A^\dagger = A$. Овакве операторе зовемо **хермитским операторима**.

Даље, посматрајмо операторе којима је хермитска конјугација једнака инверзном оператору:

$$A^\dagger = A^{-1} \qquad AA^\dagger = A^\dagger A = I$$

Овакви оператори се зову **унитарни оператори**. За њих важе нека веома важна својства, али је оно најважније да, ако је норма вектора пре извршавања била 1, и након извршавања оператора на њему, норма остаје један. Касније ће се показати зашто је ово важно[23].

2.3 Постулати квантне механике

Квантна механика може се засновати на 5 постулата. Ови постулати повезују математичку теорију са физиком, и дају потпуно нови поглед на све физичке појаве.

I постулат: Стање система

Сваки систем је у сваком тренутку описан *вектором стања* $|\psi(t)\rangle \in \mathcal{H}$, где је $\mathcal{H}$ ∞-димензиони Хилбертов простор чији вектори имају координате у $\mathbb{C}$ и чији је скуп скалара $\mathbb{C}$, а t време посматраног тренутка. Свака суперпозиција вектора стања такође је вектор стања

II постулат: Физичке величине

Свакој физичкој величини A одговара линеарни хермитски оператор $\hat{A}$, чији сопствени вектори формирају комплетну базу.

III постулат: Мерење

Мерење физичке величине A одговара примени оператора $\hat{A}$ на вектор стања $|\psi(t)\rangle$. Могуће добијене вредности мерења су сопствене вредности a_n оператора $\hat{A}$. Након мерења, систем улази у стање описано сопственим вектором ψ_n, које одговара решењу $\hat{A}\psi_n = a_n\psi_n$

IV постулат: Пробабилистички исход мерења (у дискретној бази)

При рачунању сопствене вредности a_n оператора $\hat{A}$ система у стању $|\psi\rangle$, вероватноћа добијања сопствене вредности a_n је:

$$P(a_n) = \frac{|\langle\psi_n|\psi\rangle|^2}{\langle\psi|\psi\rangle} = \frac{{a_n}^2}{\langle\psi|\psi\rangle}$$

V постулат: Временска еволуција система

Промена система, односно вектора стања система $|\psi(t)\rangle$ кроз време описана је *временски зависном Шредингеровом*[24] *једначином*

$$i\hbar\frac{\partial\,|\psi(t)\rangle}{\partial t} = \hat{H}\,|\psi(t)\rangle$$

[23] Добро је познато да све квантне капије морају представљати унитарне операторе - види [2]

[24] **Erwin Schrödinger** ***(1887-1961)*** - аустријски физичар

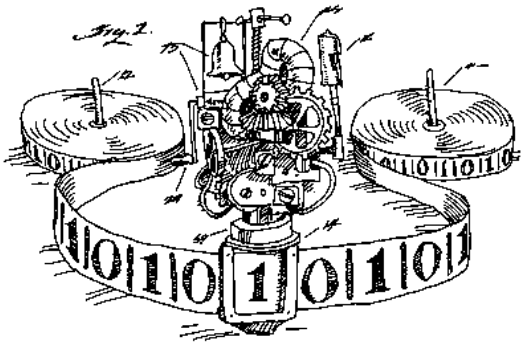
2.4 Булова алгебра. Класична логичка кола

Као што нам је већ познато, класично рачунарство је засновано на бинарном систему, над којим вршимо операције. Основна јединица података је *бит* (***bi**nary digi**t***), који представља једну бинарну цифру која може бити у стањима 0 и 1. Над битовима се врше логичке операције, и на тај начин се постиже рачунање неког одређеног жељеног резултата. Дакле, на почетку имамо систем у неком одређеном стању, који, након што се пусти кроз одређено коло, даје неко друго стање система, на основу кога вршимо даље рачунање или добијамо коначно решење.

Неке од најважнијих капија класичних рачунара су:

- **`NOT(p)`** - капија негације. За дату вредност p, враћа вредност $\neg p$
- **`AND(p,q)`** - капија конјункције. За дате вредности p и q, враћа $p \wedge q$
- **`OR(p,q)`** - капија дисјункције. За дате вредности p и q, враћа $p \vee q$
- **`XOR(p,q)`** - капија ексклузивне дисјункције. За дате вредности p и q, враћа $p \veebar q$
- **`NAND(p,q)`** - *ни*-капија. За дате вредности p и q, враћа $\neg(p \wedge q)$
- **`NOR(p,q)`** - *нили*-капија. За дате вредности p и q, враћа $\neg(p \vee q)$

Оно што је веома важно, јесте да су **`NAND`** и **`NOR`** капије универзалне. То значи да је њима могуће конструисати свако логичко коло.



Слика 3: *Тјурингова машина - уметнички приказ*

Такође, један од битних концепата рачунарства је и **Тјуринг-комплетност**. Рачунска машина је *Тјуринг-комплетна* ако је њоме могуће симулирати *Тјурингову машину*. А шта је *Тјурингова машина*? Тјурингова машина је математички модел за рачунање на апстрактној машини, који управља симболима по претходно утврђеним правилима. Машина ради на бесконачној меморијској траци, која је подељена на ћелије. Има тзв. *главу за писање*, која очитава симбол испод ње. Затим, у зависности од симбола и тренутног стања машине (тренутне инструкције):

1. уписује симбол на тренутну позицију (из коначне абецеде симбола)
2. помера се једну ћелију лево или десно
3. наставља са следећом инструкцијом или зауставља рачунање

Сви модерни рачунари су Тјуринг-комплетни.

3 Логичка кола

У свету постоји потражња за можда 5 рачунара

Томас Вотсон

Чему у ствари служи квантни компјутер? Прецизније: у чему је бољи од квантног рачунара? Томас Вотсон, директор IBM, је 1943. године рекао: *У свету можда постоји потражња за 5 рачунара*. 1945. године, у Филаделфији је направљен ENIAC[25] - први Тјуринг-комплетни електронски рачунар. У наредним годинама, рачунарство постаје једна од наука које се најбрже развијају. Данас је у употреби преко 8 милијарди рачунара - више него људи на планети.

Квантни рачунар је, теоретски, у стању да изведе неке од најважнијих задатака у драстично мањем временском интервалу - на пример, за претрагу несортираног низа дужине n би класичном рачунару било потребно $O(n)$ наредби, док квантни рачунар исти задатак ради са око $O(\sqrt{n})$ наредби[26]. Међутим, још увек нису направљени квантни рачунари довољног капацитета како би нашли реалну примену. Тренутно (април 2019.), најмоћнији квантни чип на свету - *Bristlecone* компаније Google - има 72 кубита, а тренутно највећи број факторизован квантним рачунаром је $143 = 11 \cdot 13$.

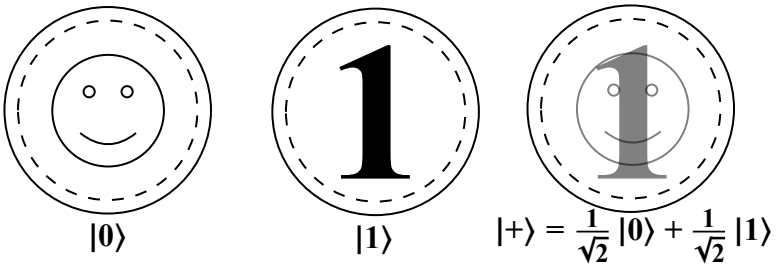
Ипак, теоријски су решени неки од најважнијих проблема данашњег рачунарства, који квантним рачунарима дају невероватну надмоћ над класичним рачунарима.

3.1 Кубити

Квантно рачунарство познаје два основна стања:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad \text{и} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Класично рачунарство и информатика засновани су на чувању информација у виду *бита* (*binary digit - bit*), који могу бити у стањима 0 и 1. Квантне информације чувају се на сличан начин. **Кубит** (***qu**antum **bit***) је свака суперпозиција стања $|0\rangle$ и $|1\rangle$ - стање: $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$.



Слика 4: *Квантни новчић*

[25] **ENIAC** - ***E**lectronic **N**umerical **I**ntegrator **a**nd **C**omputer*

[26] Види [6] и [4] за објашњење *велико O* и *мало o* нотације

Оно што разликује класични бит и квантни кубит, јесте што се мерењем бита може добити тачна информација о његовом стању - 0 или 1, док код кубита није могуће одредити тачне вредности α и β, већ искључиво $|0\rangle$ са вероватноћом $|\alpha|^2$ или $|1\rangle$ са вероватноћом $|\beta|^2$. Сходно томе, важи да је $\boldsymbol{|\alpha|^2 + |\beta|^2 = 1}$. Другачије речено, можемо рећи да је стање кубита нормирано на дужину 1 у дводимензионалном векторском простору над $\mathbb{C}$.

На пример, класичан бит можемо представити новчићем, на ком може пасти писмо (1) или глава (0) (занемаримо специјалне, готово немогуће случајеве попут тога да новчић падне на ивицу), док квантни новчић може показати стања $|0\rangle$ - глава, $|1\rangle$ - писмо, али и стања *између*, као што је $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, где је подједнака вероватноћа да је стање писмо или глава. Стања $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ и $|-\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$ ће се, због честе употребе, у даљем тексту овако означавати.

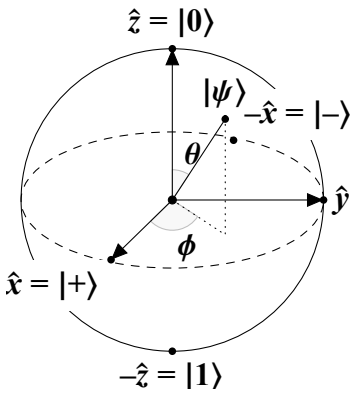
Можемо увести смене $\alpha = e^{i\gamma}\cos\frac{\theta}{2}$ и $\beta = e^{i(\gamma+\phi)}\sin\frac{\theta}{2}$. Показује се да се ово поклапа са условом $|\alpha|^2 + |\beta|^2 = 1$. Приметимо даље да се, помоћу ове замене, кубит може представити и као:

$$|\psi\rangle = e^{i\gamma}\left(\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle\right)$$

Може се показати да члан $e^{i\gamma}$ не утиче на исход мерења, односно да га можемо занемарити. Добијамо:

$$\boldsymbol{|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle}$$

што је поларна репрезентација вектора на јединичној сфери. Овим смо добили да се кубит $|\psi\rangle$ може представити као вектор на јединичној сфери - **Блоховој сфери**[27] (види слику 5).



Слика 5: *Блохова сфера*

[27] **Felix Bloch** ***(1905-1983)*** - швајцарски физичар

3.2 Вишеструки кубити

Као што класично рачунарство може посматрати више битова истовремено, и у квантном рачунарству је могуће посматрати више кубита у исто време. Постоје четири *основна стања* у којима се пар кубита може наћи - $|00\rangle$, $|01\rangle$, $|10\rangle$ и $|11\rangle$, и свака њихова суперпозиција се може представити као:

$$|\psi\rangle = \alpha_{00}\,|00\rangle + \alpha_{01}\,|01\rangle + \alpha_{10}\,|10\rangle + \alpha_{11}\,|11\rangle$$

Слично, можемо посматрати стања произвољног броја кубита $|q_1 q_2 \ldots q_k\rangle$. У том случају, на сличан начин разматрамо суперпозиције свих могућих стања. Оно што и даље важи, јесте да, редом, квадрати апсолутних вредности коефицијената представљају вероватноћу исхода свог одговарајућег вектора стања $P(|q_1 q_2 \ldots q_k\rangle) = |\alpha_{q_1 q_2 \ldots q_k}|^2$ (нпр. $P(|01\rangle) = |\alpha_{01}|^2$), те, пошто је укупна вероватноћа 1, важи:

$$\sum_{\substack{q_i \in \{0,1\} \\ (\forall i \in \{1,2,\ldots,k\})}} \left|\alpha_{q_1 q_2 \ldots q_k}\right|^2 = 1$$

Након што у систему одређеног броја кубита измеримо подскуп кубита, или на неки други начин утврдимо да су нека стања немогућа, добијамо скуп могућих стања који је подскуп почетног скупа стања. На пример, имамо суперпозицију $|\psi\rangle = \alpha_{00}\,|00\rangle + \alpha_{01}\,|01\rangle + \alpha_{10}\,|10\rangle + \alpha_{11}\,|11\rangle$ на почетку, и измеримо да први кубит има вредност 0. Тада стање $|\psi\rangle$ постаје:

$$|\psi\rangle = \frac{\alpha_{00}\,|00\rangle + \alpha_{01}\,|01\rangle}{\sqrt{|\alpha_{00}|^2 + |\alpha_{01}|^2}}$$

Једно од најважнијих оваквих стања јесте Белово стање:

$$\boldsymbol{|\beta_{00}\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}}$$

као и сродна стања $|\beta_{01}\rangle = \frac{|01\rangle+|10\rangle}{\sqrt{2}}$, $|\beta_{10}\rangle = \frac{|00\rangle-|11\rangle}{\sqrt{2}}$ и $|\beta_{11}\rangle = \frac{|01\rangle-|10\rangle}{\sqrt{2}}$ (касније ће бити више речи о овоме).

Неке од најважнијих оператора над кубитом јесу **Паулијеве**[28] **матрице** (слика 6). Ове матрице омогућавају веома корисне трансформације кубита (нпр. трансформација X представља негацију кубита - `NOT` капију), али, најбитније од свега, те четири матрице формирају потпун систем 4 базе векторског простора свих 2×2 матрица. На Блоховој сфери, матрица I представља идентитет, а матрице X, Y и Z редом представљају ротацију за π око x, y и z осе.

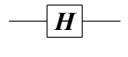
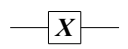
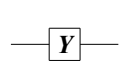
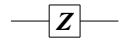
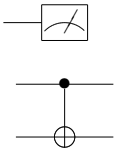
[28] **Wolfgang Pauli** ***(1900-1958)*** - швајцарски физичар

$\sigma_0 = I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\sigma_1 = X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
Нулта (*I*) Паулијева матрица	**Прва (*X*) Паулијева матрица**
$\sigma_2 = Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	$\sigma_3 = Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
Друга (*Y*) Паулијева матрица	**Трећа (*Z*) Паулијева матрица**

Слика 6: *Четири Паулијеве матрице*

3.3 Логичке капије

Логичка капија, у општем смислу, представља било који оператор над k кубита (k је произвољан број). Међутим, ради практичности, посматраћемо мањи број најважнијих капија-оператора, које се најчешће користе и са којима је могуће представити већину жељених кола, заједно са Q# кодовима[29].

Назив капије	Q# код	Графички приказ	Оператор
Адамарова капија	`H(q)`		$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
Паулијева X капија	`X(q)`		$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
Паулијева Y капија	`Y(q)`		$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$
Паулијева Z капија	`Z(q)`		$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
Капија фазног помераја θ	`Rθ(q)`		$S = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$
Мерење	`M(q)`		M
Контролисано НЕ	`CNOT(q,p)`		$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$
Замена	`SWAP(q,p)`		$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

Слика 7: *Најважније логичке капије са одговарајућим операторима*

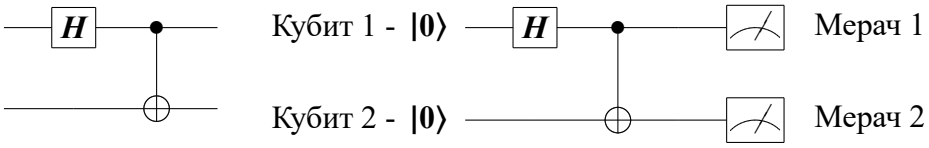
Капија *контролисано НЕ* се може представити као трансформација суперпозиције која само мења места коефицијентима уз $|10\rangle$ и $|11\rangle$:

$$\mathbf{CNOT}(|\psi\rangle) = \mathbf{CNOT}(a\,|00\rangle + b\,|01\rangle + c\,|10\rangle + d\,|11\rangle) = a\,|00\rangle + b\,|01\rangle + d\,|10\rangle + c\,|11\rangle$$

У општем случају, *контролисане капије* су капије које као *улаз* узимају један кубит који *контролише промену*, а један на ком се *извршава промена* дата неким оператором. То значи да, у случају да контролишући кубит има вредност $|0\rangle$, не долази до промене на другом кубиту (или више кубита), а уколико има вредност $|1\rangle$, долази до промене осталих кубита у складу са датим оператором (на пример, негација, ротација или замена).

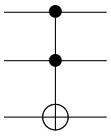
[29] Додатак А - Q# је програмски језик за симулацију квантних рачунара

На основу овога, можемо закључити да ће се управо комбинацијом ове и Адамарове капије, тзв. Беловим колом (слика 8) постизати Белово стање, колом на слици 8. Суштина овог стања је да када кубити у стању $|00\rangle$ прођу кроз Белово коло, они ће се наћи у суперпозицији $|\beta_{00}\rangle = \frac{|00\rangle+|11\rangle}{\sqrt{2}}$. Када дође до мерења на мерачу 1, доћи ће до колапса таласне функције у једно од ових стања (због III постулата квантне физике). Тада ће на мерачу 1 бити приказано 1 или 0. Кад год да дође до мерења на мерачу 2, он ће увек показати исти резултат као и мерач 1.



Слика 8: *Коло за постизање Белових стања и пример с мерачима*

Једна од веома важних капија је и **Тофолијева**[30] **капија** над 3 кубита (некада се означава као **CCNOT** - *контролисано контролисано НЕ*). Оно што издваја ову капију јесте што је њом могуће произвести свако класично логичко коло (као што то важи за функције **NAND** и **NOR**, јер $\mathbf{CCNOT}(a, b, |1\rangle) = (a, b, \mathbf{NAND}(a, b))$ за $a, b \in \{|0\rangle, |1\rangle\}$), али има примене и у квантним колима.



$$\mathbf{CCNOT} = \begin{pmatrix} \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \end{pmatrix}$$

Слика 9: *Тофолијева капија*

[30]**Tomasso Toffoli** ***(1943-)*** - италијанско-амерички физичар

4 Алгоритми засновани на појачању амплитуде

Прва група алгоритама коју проучавамо заснована је на идеји посматрања суперпозиције свих могућих решења, а затим вршења одређених трансформација којима се појачава амплитуда вектора решења. На крају, врши се мерење, при ком са веома великом вероватноћом добијамо тражено решење.

4.1 Дојч-Јожа алгоритам

Дојч[31]**-Јожа**[32] **алгоритам** је један од првих пронађених алгоритама који се на класичним рачунарима извршава у експоненцијалном времену, док се на квантном рачунару може извршити много брже (у овом случају, $O(1)$). Овај алгоритам нема значајну практичну примену, али један је од првих који показује драстичну разлику између брзине класичних и квантних рачунара.

Проблем: *Нека је дата функција* $f : \{0,1\}^n \to \{0,1\}$. *Познато је да је* f *или константна функција или балансирана функција - функција која за тачно* $1/2$ *улазних вредности даје одговор* 1*, и за тачно* $1/2$ *улазних вредности даје одговор* 0. *Одредити да ли је* f *константна или балансирана.*

На почетку, посматрамо $n+1$ кубита у стању $|\psi_0\rangle = |0\rangle^{\otimes n} |1\rangle$ (стање $|00\ldots01\rangle$ са n нула и једном јединицом на крају). Након што применимо Адамаров оператор на све кубите у $|\psi_0\rangle$, добијамо:

$$|\psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

Посматрајмо унитарни оператор $O_f : |x\rangle\,|y\rangle \to |x\rangle\,|y \oplus f(x)\rangle$. То је оператор који, примењен на број x (записан у кубитима $|x\rangle$) и *контролни кубит* $|y\rangle$ мења стање контролног кубита ако је $f(x) = 1$. Применом њега на $|\psi_1\rangle$, добијамо:

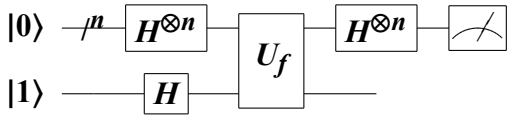
$$|\psi_2\rangle = O_f(|\psi_1\rangle) = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

Као што видимо, вредност контролног кубита се није мењала, већ се само променио знак коефицијента испред. Пошто се до краја алгоритма његова вредност неће мењати (долази само до промене знака), занемаримо овај кубит у остатку алгоритма. Даље, посматрајмо шта се дешава применом још једног Адамаровог оператора на $|\psi_2\rangle$. За сваки од могућих вектора $|x\rangle$, применом Адамаровог оператора, добијамо суперпозицију свих стања, где се за сваку јединицу у кубитском запису $|x\rangle$ мења знак испред $|x\rangle$: $|x\rangle \to \sum_{y=0}^{2^n-1} (-1)^{x\cdot y} |y\rangle$. Тиме добијамо:

$$|\psi_3\rangle = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \left(\sum_{y=0}^{2^n-1} (-1)^{x\cdot y} |y\rangle \right) = \frac{1}{2^n} \sum_{y=0}^{2^n-1} |y\rangle \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x\cdot y}$$

где је $x \cdot y = x_1y_1 \oplus x_2y_2 \oplus \ldots \oplus x_ny_n$, што је парност броја јединица у $x \wedge y$.

[31] **David Deutsch** ***(1953-)*** - британски физичар
[32] **Richard Jozsa** ***(1953-)*** - аустралијски физичар



Слика 10: *Коло Дојч-Јожа алгоритма*

Одредимо вероватноћу да при мерењу $|\psi_3\rangle$ добијемо резултат $y = 0$. Вероватноћа је једнака:

$$P_{|\psi_3\rangle}(y=0) = \left| \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x\cdot 0} \right|^2 = \left| \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \right|^2$$

Ако је $f(x)$ константна функција, уз полазну претпоставку о f, ова вероватноћа је једнака 1, а ако је балансирана, онда је једнака 0.

Дакле, ако и само ако је функција константна, биће измерена вредност 0. Ово је, стога, детерминистички алгоритам, и извршен је у $O(1)$. Класичном рачунару за ово потребно $O(2^n)$ корака, пошто би он морао да провери бар $2^{n-1}+1$ вредности функције f, како бисмо били сигурни да је функција балансирана или константна.

4.2 Гроверов алгоритам

Проблем: *Нека је дат несортиран низ са n елемената, који су индексирани (означени) природним бројевима од* 0 *до n* – 1. *Пронађи индекс на ком се појављује вредност a у низу.*

Овај проблем је опште познат под називом *претрага*. На класичном рачунару, ово решење се проналази са очекиваном сложеношћу $O(n)$. Међутим, ***Гроверов алгоритам*** нам омогућава да решење овог проблема пронађемо са сложеношћу $O(\sqrt{n})$.

Прво, поставимо кубите у стање суперпозиције свих локација, које у ствари представљају бинарне адресе података у меморији[33], помоћу n-димензионог Адамаровог оператора:

$$|\psi_0\rangle = \frac{1}{\sqrt{n}} \sum_{x=0}^{n-1} |x\rangle$$

Потребна нам је ***функција квантног одлучивања*** - црна кутија[34] која има могућност да *препозна* решење овог проблема. Ово не значи да она аутоматски може *пронаћи* решење, већ да може да за неки број *препозна* да ли је решење проблема. Ако је a тражени број, функција квантног одлучивања се, у ствари, може представити као функција:

$$p(x) = \begin{cases} 1 & \text{ако је } f(x) = a \\ 0 & \text{иначе} \end{cases}$$

[33] Локације се могу уопштено посматрати као елементи домена функције $f: \{0, 1, ..., n-1\} \to CD$, за произвољни кодомен, који је обично скуп целих или реалних бројева. Тада, Гроверов алгоритам за дати елемент кодомена y тражи одређени елемент домена x који се пресликава у y: $f(x) = y$. У практичним случајевима, функција f обично представља меморију рачунара, док кодомен чине вредности уписане у меморију. Види [5]

[34] *Црном кутијом* се обично називају елементи (квантног кола у овом случају) за које нам није неопходно знати детаљан начин рада, обично јер није од претераног значаја за даље разматрање проблема који посматрамо.

У ствари, можемо дефинисати оператор $O_p : |x\rangle\,|c\rangle \rightarrow |x\rangle\,|c \otimes p(x)\rangle$ - оператор *квантног одлучивања*. То је оператор који ће *обрнути* стање (променити у супротно стање) *контролног кубита* $|c\rangle$, уколико је тај кубит једнак траженом броју. Слично као код Дојч-Јожа алгоритма, уколико се контролни кубит налазио у стању $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$, доћи ће до следеће трансформације (посматрамо за једну вредност $x \in \{0, 1, \ldots, n\}$):

$$|\chi_1\rangle = |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) \rightarrow |\chi_2\rangle = O_p\,|\chi_1\rangle = (-1)^{p(x)}\,|x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

Пошто се стање последњег кубита касније неће мењати током целог алгоритма, можемо га надаље слободно занемарити ($|\chi\rangle = |\phi\rangle \otimes |c\rangle$). Тада имамо оператор: $O_p : |x\rangle \rightarrow (-1)^{p(x)}\,|x\rangle$. O_p можемо представити као $O_p = I - 2\,|p\rangle\,\langle p|$.

$$|\phi_1\rangle = |x\rangle \rightarrow |\phi_2\rangle = O_p\,|\phi_1\rangle = (-1)^{p(x)}\,|x\rangle$$

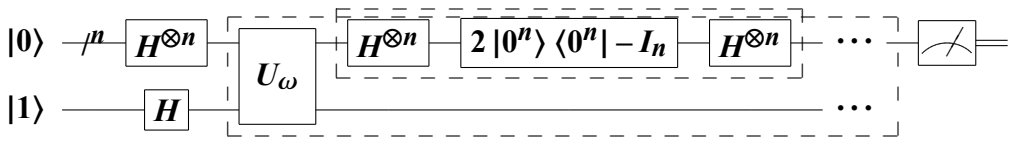
Даље, нека је $|s\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} |k\rangle = |\psi_0\rangle$ суперпозиција свих могућих локација са једнаком вероватноћом за сваку локацију, и нека је решење алгоритма које тражимо $|m\rangle$. Тада важи $O_p = I - 2\,|m\rangle\,\langle m|$. Посматрајмо сада оператор $O_s = 2\,|s\rangle\,\langle s| - I$. Након што применимо O_p на вектор

$$|\psi_2\rangle = \sum |\phi_2\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} (-1)^{p(x)}\,|x\rangle$$

добијамо нови вектор, код ког је, у суштини, повећан коефицијент уз $|m\rangle$, а благо равномерно умањен уз све остале чланове.

$$\begin{aligned} |\psi_3\rangle &= O_s\,|\psi_2\rangle \\ &= O_s O_p\,|s\rangle \\ &= (2\,|s\rangle\,\langle s| - I)(I - 2\,|m\rangle\,\langle m|)\,|s\rangle \\ &= \frac{n-4}{n}\,|s\rangle + \frac{2}{\sqrt{n}}\,|m\rangle \end{aligned}$$

Гроверов оператор $\boldsymbol{O_s = 2\,|s\rangle\,\langle s| - I}$



Поновити $\boldsymbol{O(\sqrt{N})}$ пута

Слика 11: *Коло Гроверовог алгоритма*

Посматрајмо сада дводимензионални векторски простор одређен елементарним векторима $|m\rangle$ и $|r\rangle = \sqrt{\frac{n}{n-1}}\,|s\rangle - \frac{1}{\sqrt{n}}\,|m\rangle$, где је $|m\rangle$ тражено решење, а $|r\rangle$ суперпозиција свих осталих индекса

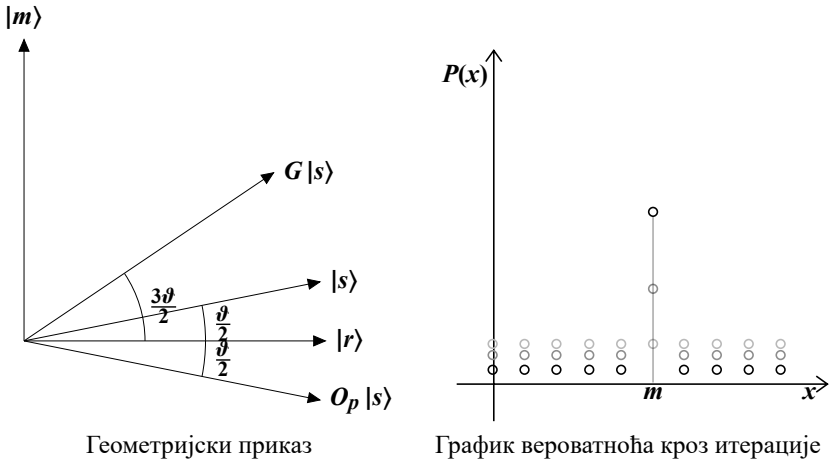
са једнаком вероватноћом. Иницијални вектор $|s\rangle$ налазио се под неким углом $\frac{\vartheta}{2}$ у односу на $|r\rangle$. Посматрајмо шта ће се десити применом оператора O_p и O_s.

Прво, по примени оператора O_p долази само до промене знака испред $|m\rangle$, што је еквивалентно рефлексији у односу $|r\rangle$, док примена оператора O_s рефлектује тренутни вектор стања у односу на $|s\rangle$. Као композиција две рефлексије, оператор $O_s O_p = G$ (тзв. *Гроверов оператор*) представља ротацију за угао ϑ у позитивном смеру. Дакле:

$$G^k |s\rangle = \cos\left(\frac{2k+1}{2}\vartheta\right)|m\rangle + \sin\left(\frac{2k+1}{2}\vartheta\right)|r\rangle$$

Сходно томе, може се закључити да се овај оператор треба извршити довољно пута да би важило $\frac{2k+1}{2}\vartheta \approx \frac{\pi}{2}$, односно $k \approx \frac{\pi}{2\vartheta} - \frac{1}{2}$. Пошто је $\frac{\vartheta}{2}$ угао између суперпозиције свих стања $|s\rangle$ и суперпозиције свих стања сем $|m\rangle$ - стање $|s\rangle$, а притом важи $|s\rangle = \sqrt{\frac{N-1}{N}}|r\rangle + \frac{1}{\sqrt{N}}|m\rangle$, односно $\sin\vartheta = 2\sin\left(\frac{\vartheta}{2}\right)\cos\left(\frac{\vartheta}{2}\right) = 2\sqrt{\frac{1}{N}}\sqrt{\frac{N-1}{N}} = \frac{2\sqrt{N-1}}{N}$, па треба да важи да је: $k \approx \frac{N\pi}{4\sqrt{N-1}} + \frac{1}{2} \approx \frac{\pi\sqrt{N}}{4}$. Сходно томе, узимамо:

$$k = \left\lceil \frac{\pi\sqrt{2^n}}{4} \right\rceil$$



Геометријски приказ — График вероватноћа кроз итерације

Слика 12: *Графички прикази Гроверовог алгоритма*

Дакле, кораци алгоритма су:

1. **Иницијализација кубита у стање $|s\rangle$**
2. **Циклус приближавања решењу, извршава се $\left\lceil \frac{\pi\sqrt{2^n}}{4} \right\rceil$ пута**
   - **(а) Примена оператора O_p**
   - **(б) Примена оператора O_s**
3. **Мерење резултата**

5 Алгоритми са кв. Фуријеовом трансформацијом

Фуријеова трансформација је једна од најважнијих математичких појмова за анализу функција. Ретки су случајеви трансформација које су толико важне да се потпуно засебно изучавају. Једно од кључних открића квантног рачунарства је да се неке од оваквих трансформација много брже рачунају на квантном компјутеру. Самим тим, отворен је велики број могућности за убрзавање и побољшање рачунских процеса у односу на класичне рачунаре.

5.1 Квантна Фуријеова трансформација

Фуријеова трансформација *разбија* функцију на периодичне синусоидалне компоненте од којих се она састоји. Ако означимо да је $N = 2^n$ и $\varepsilon_n = e^{\frac{2\pi i}{N}}$ нетривијалан n-ти корен броја 1, за сваки улазни вектор комплексних бројева $(x_0, x_1, \ldots, x_{N-1}) \in \mathbb{C}^n$, можемо одредити вектор $(y_0, y_1, \ldots, y_{N-1})$:

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \cdot {\varepsilon_n}^{-jk}$$

На сличан начин функционише и ***квантна Фуријеова трансформација***. Ако је дато стање $|j\rangle$, и ако је опсег могућих вредности датог низа кубита од 0 до n, квантна Фуријеова трансформација стања $|j\rangle$ је:

$$|j\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} |k\rangle \cdot {\varepsilon_n}^{jk}$$

На основу овога можемо закључити да је Фуријеова трансформација на произвољно стање:

$$F_n : \sum_{j=0}^{N-1} x_j \, |j\rangle \rightarrow \sum_{k=0}^{N-1} y_k \, |k\rangle$$

F_n се може представити и као матрица (слика 13). F_n је унитарна трансформација, што се лако показује провером да важи $F_n {F_n}^\dagger = {F_n}^\dagger F_n = I$.

$$\boldsymbol{F_n} = \frac{\mathbf{1}}{\sqrt{\boldsymbol{n}}} \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \cdots & \mathbf{1} \\ \mathbf{1} & \boldsymbol{\varepsilon_n} & {\boldsymbol{\varepsilon_n}}^{\mathbf{2}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{3}} & \cdots & {\boldsymbol{\varepsilon_n}}^{\boldsymbol{N-1}} \\ \mathbf{1} & {\boldsymbol{\varepsilon_n}}^{\mathbf{2}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{4}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{6}} & \cdots & {\boldsymbol{\varepsilon_n}}^{\mathbf{2(N-1)}} \\ \mathbf{1} & {\boldsymbol{\varepsilon_n}}^{\mathbf{3}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{6}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{9}} & \cdots & {\boldsymbol{\varepsilon_n}}^{\mathbf{3(N-1)}} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{1} & {\boldsymbol{\varepsilon_n}}^{\boldsymbol{N-1}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{2(N-1)}} & {\boldsymbol{\varepsilon_n}}^{\mathbf{3(N-1)}} & \cdots & {\boldsymbol{\varepsilon_n}}^{\mathbf{(N-1)(N-1)}} \end{pmatrix}$$

Слика 13: *Матрица за квантну Фуријеову трансформацију*

Пошто су све квантне операције линеарне (а F_n је унитарна матрица, а самим тим и валидна квантна операција), можемо посматрати резултате трансформација појединачних основних стања, а свака суперпозиција стања ће се моћи одредити као линеарна комбинација резултата основних стања. Нека су $k_1, k_2, \ldots, k_n$ цифре одговарајућег броја k. Посматрајмо резултат Фуријеове трансформације броја $|j\rangle$:

$$\begin{aligned}
|j\rangle &\to \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{jk\frac{2\pi i}{2^n}} \\
&= \frac{1}{2^{n/2}} \sum_{k_1=0}^{1} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} e^{2\pi ij\left(\sum_{l=1}^{n} k_l 2^{-l}\right)} |k_1 k_2 \ldots k_n\rangle \\
&= \frac{1}{2^{n/2}} \sum_{k_1=0}^{1} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} \bigotimes_{l=1}^{n} e^{2\pi ijk_l 2^{-l}} |k_l\rangle \\
&= \frac{1}{2^{n/2}} \bigotimes_{l=1}^{n} \sum_{k_l=0}^{1} e^{2\pi ijk_l 2^{-l}} |k_l\rangle \\
&= \frac{1}{2^{n/2}} \bigotimes_{l=1}^{n} \left(|0\rangle + e^{2\pi ij2^{-l}} |1\rangle\right)
\end{aligned}$$

Посматрајмо сада део израза $e^{2\pi ij2^{-l}}$. Нека је број $j = [j_1 j_2 \ldots j_n]$ записан у бинарном систему. Тада важи:

$$e^{2\pi ij2^{-l}} = e^{2\pi i[j_1 j_2 .. j_n]2^{-l}} = e^{2\pi i[j_1 j_2 .. j_l . j_{l+1} .. j_n]} = e^{2\pi i[0.j_{l+1} .. j_n]}$$

Дакле, Фуријеова трансформација трансформише број $|j_1 j_2 \ldots j_n\rangle$ на следећи начин:

$$|j_1 j_2 \ldots j_n\rangle \to \frac{1}{2^{n/2}} \bigotimes_{l=0}^{n} \left(|0\rangle + e^{2\pi i[0.j_{l+1} .. j_n]} |1\rangle\right)$$

Како конструисати коло за ову трансформацију? Овај проблем је лакши него што се чини. Посматрајмо прво шта се деси ако применимо Адамарову трансформацију на кубит $|j_1\rangle$:

$$H(|j_1\rangle) = \frac{1}{\sqrt{2}}\left(|0\rangle + (-1)^{j_1} |1\rangle\right) = \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i[0.j_1]} |1\rangle\right)$$

$$|j\rangle = |j_1 j_2 \ldots j_n\rangle \to \left(|0\rangle + e^{2\pi i[0.j_1]} |1\rangle\right) \otimes |j_2 \ldots j_n\rangle$$

Нека је трансформација $R_n(x)$ ротација за угао $\frac{2\pi}{2^n}$. Контролисана ротација $CR_n(x, y)$ је трансформација која помера фазу кубита x само за $|1\rangle$ компоненту кубита y, односно

$$CR_n(x, y) = \langle y|0\rangle |x\rangle \otimes |y\rangle + \langle y|1\rangle R_n(|x\rangle) \otimes |y\rangle$$

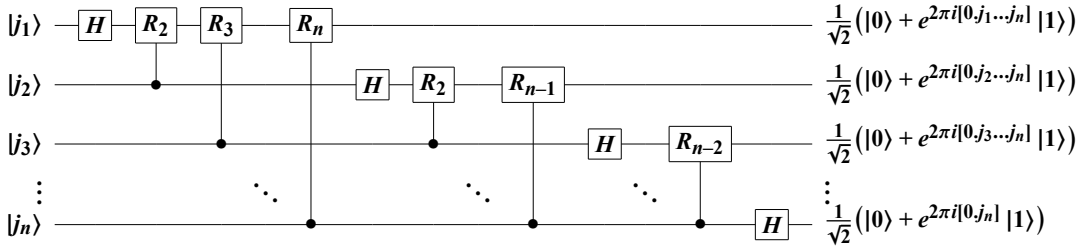
Ако применимо трансформацију R_2 контролисану кубитом $|j_2\rangle$ на трансформисаном кубиту $H(|j_1\rangle)$, добијамо:

$$H(|j_1\rangle) = \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i[0.j_1]} |1\rangle\right) \xrightarrow{CR_2} \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i[0.j_1]} e^{2\pi i\frac{j_2}{2^2}} |1\rangle\right) = \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i[0.j_1 j_2]} |1\rangle\right)$$

На сличан начин, настављамо да примењујемо редом R_3, R_4, ..., R_n контролисане кубитима $|j_3\rangle$, $|j_4\rangle$, ..., $|j_n\rangle$, редом, чиме добијамо:

$$|j_1\rangle \to \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i[0.j_1 j_2 \ldots j_n]}\,|1\rangle\right)$$

Слично радимо и за остале кубите - на кубит $|j_k\rangle$ применимо прво Адамарову трансформацију H, а затим редом примењујемо R_2, ..., R_{n+1-k} контролисане кубитима $|j_{k+1}\rangle$, ..., $|j_n\rangle$. На крају заиста добијамо F_n, чије коло изгледа као на слици 14.



Слика 14: *Коло квантне Фуријеове трансформације* $\boldsymbol{F_n}$

5.2 Процена фазе

Као што смо видели, *квантна Фуријеова трансформација* нам, за датих n цифара, даје n стања суперпозиције са једнаким вероватноћама за $|0\rangle$ и $|1\rangle$, чија је фазна разлика једнака $[0.j_k j_{k+1} \ldots j_n]$ - последњих $n-k$ цифара се распоређује *иза зареза*. Дакле, ми као резултат Фуријеове трансформације добијамо (релативно) прецизне познате фазе кубита у суперпозицији. Пошто је свака капија реверзибилна, ми можемо као *улаз* кола дати управо ове фазно померене суперпозиције, и одредити њихове фазе са прецизношћу од n цифара. Међутим, ово важи само ако фазе деле *наставке* цифара (као што је то на слици 14), односно, ако је свака следећа фаза квадрат претходне.

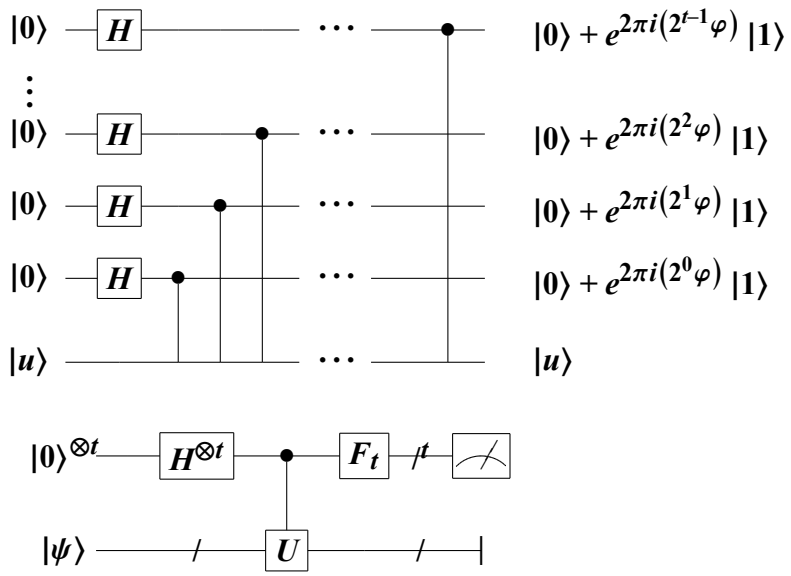
Уколико фаза настаје као последица примене неке унитарне трансформације U, можемо ту трансформацију применити још једном, чиме ћемо добити квадрат дате фазе. Ако применимо $U\,2^k$ пута, добићемо фазу померену за k бинарних цифара улево. На овај начин можемо конструисати свих n стања резултата квантне Фуријеове трансформације. Како је квантна Фуријеова трансформација реверзибилна, можемо одредити колика је та фаза.

Процена фазе φ сопственог вектора $|u\rangle$ унитарне матрице U (сопствена вредност вектора $|u\rangle$ је $e^{2\pi i\varphi}$) је претходно описани поступак, који ћемо сада детаљно описати. При томе, претпостављамо да су нам доступне *црне кутије* за припрему стања $|u\rangle$ и за извршавање контролисане операције U^{2^k} за $k \in \mathbb{N}_0$.

На почетку имамо два регистра - први регистар са t кубита (t зависи од тражене прецизности решења - вероватноће добијања тачног и броја цифара које желимо да добијемо), који је иницијализован на $|0\rangle^{\otimes n}$, и други регистар у коме је уписан $|u\rangle$ са довољним бројем кубита.

Лако се може видети да као резултат рада првог дела кола добијамо:

$$|\psi_0\rangle = |0\rangle \to |\psi_1\rangle = \frac{1}{2^{n/2}}\left(|0\rangle + e^{2\pi i 2^{t-1}\varphi}\,|1\rangle\right)\ldots\left(|0\rangle + e^{2\pi i 2^{0}\varphi}\,|1\rangle\right) = \frac{1}{2^{n/2}}\sum_{k=0}^{2^n-1} e^{2\pi i k\varphi}\,|k\rangle$$



Слика 15: *Приказ кола за ӣрви део одређивања фазе и уӣрошћеноӣ кола за одређивање фазе*

Оно што се такође може приметити, јесте да ако представимо фазу као $\varphi = [0.\varphi_1\varphi_2 \ldots \varphi_t]$, тада је:

$$|\psi_1\rangle = \frac{1}{2^{n/2}}\left(|0\rangle + e^{2\pi i[0.\varphi_t]i}\,|1\rangle\right)\left(|0\rangle + e^{2\pi i[0.\varphi_{t-1}\varphi_t]}\,|1\rangle\right)\ldots\left(|0\rangle + e^{2\pi i[0.\varphi_1\varphi_2\ldots\varphi_t]}\,|1\rangle\right)$$

Применом инверзне Фуријеове трансформације, добијамо тражене бројеве. Ипак, шта се дешава ако φ није тачно једнако $[0.\varphi_1\varphi_2 \ldots \varphi_t]$? Колика је грешка мерења, ако је оно могуће?

Може се показати да је овако добијено $|\phi\rangle$ прилично добра апроксимација за $|\varphi\rangle$. Уколико желимо да израчунамо $|\varphi\rangle$ са прецизношћу од n цифара, и вероватноћом успеха $1-\epsilon$, требало би узети[35]:

$$t = n + \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil$$

5.3 Ред броја по модулу и Шоров алгоритам

Познато је да је ред броја x по модулу n је најмањи број ρ већи од 2 тако да је $x^\rho \equiv 1 \pmod{n}$. Посматрајмо унитарни оператор:

$$U\,|y\rangle \equiv |xy \pmod{N}\rangle$$

Такође, сматрамо да за $y \geq N$ важи $U\,|y\rangle = |y\rangle$. Ако задржимо ове ознаке, једноставним рачуном показујемо да су, за $0 \leq s \leq \rho$, следећа стања сопствени вектори U:

$$|u_s\rangle = \frac{1}{\sqrt{r}}\sum_{k=0}^{r-1} e^{\frac{2\pi isk}{r}}\left|x^k \bmod N\right\rangle$$

[35]Ово тврђење је изведено у књизи [5]

Тада су $e^{\frac{2\pi is}{r}}$ одговарајуће сопствене вредности оператора U, пошто важи:

$$|u_s\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{\frac{2\pi isk}{r}} \left| x^{k+1} \bmod N \right\rangle$$
$$= e^{\frac{2\pi is}{r}} |u_s\rangle$$

Уколико применимо алгоритам одређивања фазе квантном Фуријеовом трансформацијом на унитарни оператор U, са добром прецизношћу добијамо фазу $e^{\frac{2\pi is}{r}}$, одакле лако можемо одредити r. Посматрајмо сада следеће две теореме:

Теорема 5.1 *Нека је* N *сложен број (записан помоћу* L *битова), и нека је* x *нетривијално решење једначине* $x^2 \equiv 1 \pmod N$ *- за које важи* $x \not\equiv \pm 1 \pmod N$ *и* $1 \leq x \leq N$*. Тада је бар један од бројева* $NZD(x-1, N)$ *и* $NZD(x+1, N)$ *нетривијални фактор броја* N *који можемо рачунати у* $O(L^3)$

Теорема 5.2 *Нека је* $N = p_1^{\alpha_1} \ldots p_m^{\alpha_m}$ *факторизација непарног сложеног броја. Нека је* x *униформно насумично одабрани број, за који важи* $1 \leq x \leq N$ *и* $NZD(x, N) = 1$*. Нека је ред броја* x *по модулу* n*. Тада важи:*

$$P(2|r \wedge x^{r/2} \not\equiv -1 \pmod N) \geq 1 - \frac{1}{2^m}$$

Шта ово значи? Прва теорема нам говори да лако можемо да од нетривијалног броја реда 2 пронађи неки делилац броја N, док друга теорема говори да је велика вероватноћа *погађања* нетривијалног броја који има ред 2 по модулу N. Сходно томе, применом следећег алгоритма - **Шоровог**[36] **алгоритма** - можемо брзо факторизовати произвољан природан број.

Једини преостали проблем је мала вероватноћа одабира нетривијалног x у случају $m = 1$. У кратком времену $O((\log N)^2)$ се може проверити да ниједан од $\sqrt[k]{N}$ за $k \leq \log_2(N)$ није природан број (ако јесте, онда је то решење).

Дакле, поступак Шоровог алгоритма је:

1. **Ако је број паран, врати 2**
2. **Ако постоји $b \leq \log_2(N)$ тако да $a^b = N$, врати a**
3. **Одабери насумично $1 \leq k \leq N-1$. Ако је $NZD(x, N) > 1$, врати $NZD(x, N)$**
4. **Пронађи ред ρ броја k по модулу N**
5. **Ако је r паран и $x^{r/2} \not\equiv -1 \pmod N$, одреди $NZD(x-1, N)$ и $NZD(x+1, N)$, и ако су неки од њих нетривијални фактори врати их као резултат**
6. **Ако нема решења, врати се на корак 3**

Очекивана брзина овог недетерминистичког алгоритма је $O((\log n)^3)$. Постоје оптимизације овог алгоритма до $O((\log n)^2(\log\log n)(\log\log\log n))$, али су обе варијанте, у сваком случају, далеко брже од најбржег класичног алгоритма, који ради у $O(e^{1.9(\log N)^{1/3}(\log\log N)^{2/3}})$.

[36] **Peter Shor** ***(1959-)*** - амерички математичар

6 Програмски језик Q#

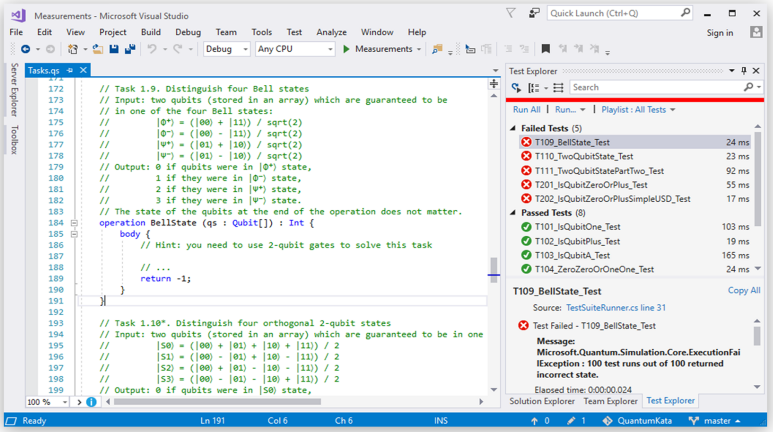
Кад говоримо о атомима и квантној физици, језик се може користити само као поезија

Нилс Бор

11. децембра 2017. године, *Microsoft* корпорација је објавила нови производ *Quantum Development Kit* који служи за развој квантних алгоритама, у ком је укључен нови програмски језик Q# (*кју-шарп*). Између осталог, овај програм се може користити и за симулацију рада квантног рачунара на класичном рачунару. Иако је класичан рачунар машина која није способна за прави пробабилистички рачун, као ни приближну брзину квантног рачунара, добре псеудослучајне функције и оптимизовани симулирани кубити омогућују преглед неких једноставнијих алгоритама, као и поприлично уверљиву симулацију рада с кубитима. У овом додатку, наводе се примери кодова који симулирају поједина кола квантних рачунара.

6.1 Синтакса Q#

Програмски језик Q# заснован је на програмским језицима C# и F#, и велики број карактеристика је наслеђено од ових језика. Q# је, у ствари, програмски језик у ком се програмирају искључиво квантни оператори (*операције* - кључна реч `operations`) и функције (`functions`). Користе се типови података слични онима у C#, само са великим почетним словима: `Int`, `Bool`, `Double` и `String`. Кубити се алоцирају и касније одбацују унутар `using` блока, повратне вредности се враћају помоћу `return`, а ламбда функције додељујемо помоћу `=>` Такође, слично програмском језику F#, промењиве се декларишу кључном речи `let` или `mutable`, *for* циклус се дефинише као `for ... in`, а `..` је оператор опсега. Остатак документације се може видети у [9] и [10], а даље ћемо видети начин функционисања језика кроз примере.



Слика 16: *Развојно окружење за Q# - Visual Studio*

6.2 Белово стање

Као што смо већ помињали, коло у ком кубит прво пролази кроз Адамарову, а затим, заједно с другим кубитом, кроз **CNOT** капију (слика 8), назива се *Белово коло*. Ако као почетне кубите задамо $|00\rangle$, односно $|11\rangle$, требало би да добијемо да при мерењу, оба кубита увек дају исти резултат, јер ће се налазити у стању $\frac{|00\rangle+|11\rangle}{\sqrt{2}}$, односно $\frac{|00\rangle-|11\rangle}{\sqrt{2}}$, па ће бити у стању у ком су сигурно једнаки, са вероватноћом око 50% за $|00\rangle$ или $|11\rangle$.

Код који симулира дато коло је:

```
operation Postavi (rez: Result, poc: Qubit) : Unit {
    let tmp = M(poc);
    if (rez != tmp) {
        X(poc);
    }
}

operation BelovaStanja (ukupno : Int, poc: Result) : (Int, Int, Int)
{
    mutable jedinice = 0;
    mutable podudaranja = 0;
    using (kubiti = Qubit[2]) {
        for (test in 1..ukupno) {
            Postavi (poc, kubiti[0]);
            Postavi (Zero, kubiti[1]);
            H(kubiti[0]);
            CNOT(kubiti[0], kubiti[1]);
            let res = M (kubiti[0]);

            if (M (kubiti[1]) == res) {
                 set podudaranja = podudaranja + 1;
             }

            if (res == One) {
                set jedinice = jedinice + 1;
            }
        }
        Postavi(Zero, kubiti[0]);
        Postavi(Zero, kubiti[1]);
    }
    return (ukupno-jedinice, jedinice, podudaranja);
}
```

Резултат извршавања операције за **ukupno = 1000** и **poc = <Поч. вредност>** је:

```
Поч. вредност  |0>  |1>   Подударања
Zero (|0>)     503  497   1000
One (|1>)      506  494   1000
```

У 50.45% случајева измерен је кубит $|0\rangle$, а у 49.55% случајева кубит $|1\rangle$. Као што видимо, резултат се понаша у складу с очекиваним вредностима од 50%. Такође, у свих 1000 случајева је дошло до поклапања.

6.3 Гроверово коло

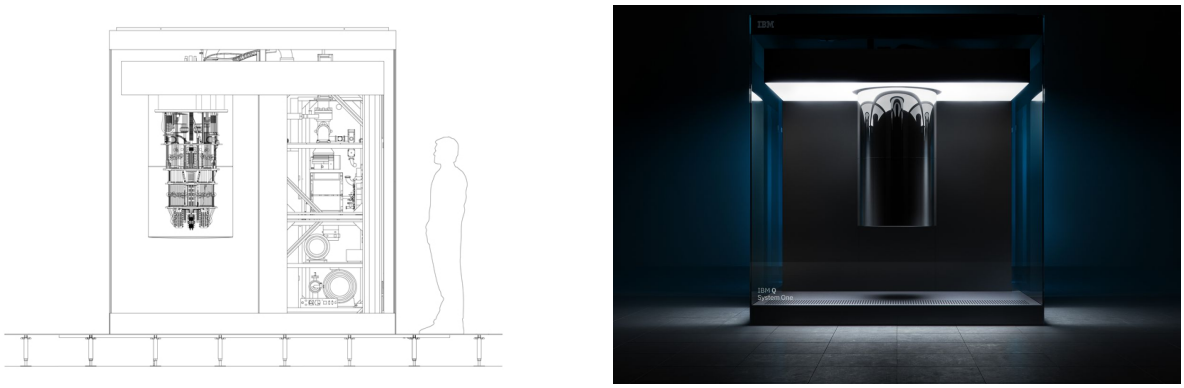
Напишимо сад код за симулацију Гроверовог кола (слика 11). Као што смо већ помињали, при извршавању Гроверовог алгоритма потребно је прво извршити припрему стања, тако што ће се регистар локације иницијализовати на вредност униформне суперпозиције свих локација. Након тога, *маркирамо* тражено решење применом црне кутије - променимо знак контролног кубита за локацију решења. Сада треба још да $\lceil 2^{n/2}\pi/4 \rceil$ пута применимо Гроверов алгоритам појачавања амплитуде - рефлексија око $|s\rangle$ (рефлексија око 0, угњеждена између Адамарових трансформација) и рефлексија око $|r\rangle$ (промена знака маркираног). Следећи код извршава овај алгоритам:

```
operation CrnaKutija (kontKubit : Qubit, registar : Qubit[]) :
                                    Unit is Adj + Ctl {
    // Controlled X je kapija CNOT
    Controlled X(registar, kontKubit);
}
operation UniformnaSuperpozicija (registar : Qubit[]) : Unit is Adj + Ctl {
    ApplyToEachCA(H, registar);
}
operation PripremaStanja (kontKubit : Qubit, registar : Qubit[]) :
                                    Unit is Adj + Ctl {
    UniformnaSuperpozicija(registar);
    CrnaKutija(kontKubit, registar);
}
operation RefleksijaMarkirani (kontKubit : Qubit) : Unit {
    R1(PI(), kontKubit);
}
operation RefleksijaNula (registar : Qubit[]) : Unit {
    ApplyToEachCA(X, registar);
    Controlled Z(Rest(registar), Head(registar));
    ApplyToEachCA(X, registar);
}
operation RefleksijaS (kontKubit : Qubit, registar : Qubit[]) : Unit {
    Adjoint PripremaStanja(kontKubit, registar);
    RefleksijaNula([kontKubit] + registar);
    PripremaStanja(kontKubit, registar);
}
operation QPretraga (brIteracija : Int, kontKubit : Qubit,
                                    registar : Qubit[]) : Unit {
    PripremaStanja(kontKubit, registar);
    // Groverove iteracije
    for (idx in 0 .. brIteracija - 1) {
        RefleksijaMarkirani(kontKubit);
        RefleksijaS(kontKubit, registar);
    }
}
```

7 Закључак

Као што смо могли да видимо, квантно рачунарство је релативно нова наука. Иако смо открили и теоријски засновали неке веома важне појмове и алгоритме, и дали веома значајне примене, дугачак пут је пред нама. Још увек смо веома далеко од свакодневне комерцијалне употребе, и биће потребно још много времена, истраживања и размишљања како бисмо заиста применили ову науку у свакодневном животу. Ипак, сваког дана изађе по нека вест о новом пробоју у квантној физици, и стално се нешто ново појављује и открива.

Квантни рачунари су много бољи од класичних у решавању одређених проблема. С обзиром да се сва класична кола могу симулирати на њима, они могу извршити све алгоритме које могу и класични рачунари са, у најгорем случају, истом брзином. Поред тога, постоје одређени алгоритми који дају драстична побољшања брзине у односу на класичне рачунаре, као што су Гроверов алгоритам за претрагу ($O(n) \rightarrow O(\sqrt{n})$), Шоров алгоритам факторизације природних бројева ($O(e^{1.9 \log n \log\log n}) \rightarrow O((\log n)^2(\log\log n)(\log\log\log n))$) или квантни алгоритам за решавање система једначина ($O(n\kappa) \rightarrow O(\log n\kappa^2$)[37])



Слика 17: *Први комерцијални квантни рачунар - IBM Q System One*

Иако тренутно најмоћнији квантни рачунар на свету има само 72 кубита, и иако смо до сада факторизовали само број 143, квантни рачунари се све брже и брже развијају. IBM је најавио пуштање првог правог квантног рачунара (*IBM Q System One* - слика 17) са 20 кубита у продају, а у развоју је и комерцијални рачунар са чак 58 кубита. Неколико других компјутерских гиганата, као што су *Google*, *Microsoft*, *Alibaba*, *Fujitsu* и *NEC* такође има сличне пројекте. Моћ квантних рачунара расте, цена опада, технологија се развија, а знања је све више. Будућност ове науке ће бити веома узбудљива.

[37] κ број који означава *осетљивост* функције - мера промене функције у зависности од мале промене њених параметара

Ову тему сам одабрао зато што сам очекивао веома занимљиву теорију, која би могла имати невероватну примену. Ипак, нисам могао да претпоставим колико ме је истраживање на ову тему интригирало, одушевило и заинтересовало. Иако су у овом раду обрађена само два најпознатија алгоритма, постоје многи други проблеми, као и приступи њиховом решењу, са још невероватнијим идејама и концептима, као што је проблем проналажења троугла у графу или постојања истих елемената у скупу. Квантно рачунарство убрзава решавање неких од највећих проблема данашњице, и највећа открића су тек пред нама. Коначно, желео бих да одговорим на питање из дела 1.1: Можемо ли даље? Наравно да можемо!

На крају, желео бих да изразим велику захвалност свом ментору, професору Игору Салому, на инспирацији и помоћи при истраживању и изради овог рада. Такође, велику захвалност дугујем и свим својим професорима, а посебно Соњи Чукић и Наташи Чалуковић, на вишегодишњем труду, залагању, помоћи и мотивацији за рад и успех.

Литература

[1] P. Ceruzzi, ***A History of Modern Computing (2nd Edition)***, MIT Press, 2003.

[2] N. Zettili, ***Quantum Mechanics - Concepts and applications (2nd Edition)***, Wiley, 2009.

[3] Н. Чалуковић, ***Физика за 4. разред гимназије***, Круг, 2014.

[4] З. Каделбург, В. Мићић, С. Огњановић, ***Анализа с алгебром за 4. разред гимназије***, Круг, 2016.

[5] M. Nielsen, I. Chuang, ***Quantum Computation and Quantum Information (10th Anniversary Ed.)***, Cambridge University Press, 2011.

[6] D. Knuth, ***Big Omicron and Big Omega and Big Theta***, Stanford University, 1976.

[7] A. Harrow, A. Hassidim, S. Lloyd, ***Quantum algorithm for linear systems of equations***, Physical Review Letters, 2009.

[8] А. Харт-Дејвис, ***Наука - велика илустрована енциклопедија***, Младинска књига, 2011.

[9] Wikimedia фондација, ***Википедија, слободна енциклопедија***, www.wikipedia.org

[10] Microsoft корпорација, ***Документација програмског језика Q#***, docs.microsoft.com/en-us/quantum