

**PEMBANGUNAN INDEKS KEKOMPLEKSAN
BAHASA C BERASASKAN ASPEK KUALITATIF
SECARA KAJIAN EMPIRIKAL**

Aziz Deraman
Fakulti Teknologi dan Sains Maklumat,
Universiti Kebangsaan Malaysia,
43600 Bangi, MALAYSIA
email: ad@ftsm.ukm.my

ABSTRACT

There are a number of techniques that could be used to measure program difficulty level such as Halstead software science, McCabe and McClure. However, these are mostly limited to counting programs' operands and operators. The qualitative aspect is often neglected. The qualitative aspect is to include the background, environment, working culture, academic level, social and cultural aspect that influence the developers' view of what aspects of programming that is considered as 'complex'. This paper presents findings from an empirical study for the development of C language complexity index with local qualitative features. With the newly developed C language complexity index, the measurement of related metrics will be more accurate and reliable.

ABSTRAK

Kini terdapat banyak kaedah yang boleh digunakan untuk mengukur paras kesukaran atur cara seperti sains perisian Halstead, McCabe dan McClure. Bagaimana pun, kesemua pengukuran ini hanya terhad kepada penghitungan statik kendalian dan pengendali yang terdapat di dalam atur cara. Dalam hal ini aspek kualitatif biasanya dilupakan. Di antara aspek kualitatif yang penting berkaitan dengan latar belakang, persekitaran, budaya kerja, tahap akademik, sosial dan budaya. Aspek-aspek ini sebenarnya telah mempengaruhi pandangan penulis atur cara mengenai aspek pengaturcaraan yang dianggap kompleks. Kertas ini membentangkan hasil satu kajian empirikal untuk membina indeks kekompleksan bahasa C mengikut ciri kualitatif tempatan. Dengan indeks ini diharapkan proses pengukuran kekompleksan atur cara C untuk pelbagai tujuan menjadi lebih tepat dan boleh dipercayai.

Pengenalan

Kini perisian telah menjadi faktor utama sebagai penentu kejayaan bagi kebanyakan sistem yang berasaskan komputer berbanding dengan aspek perkakasan. Perisian merupakan faktor yang membezakan antara komputer yang digunakan untuk menjalankan perniagaan, komputer yang digunakan untuk mengawal robot serta komputer yang digunakan untuk membantu proses pembelajaran. Ketepatan dan kekinian sesuatu maklumat yang dihasilkan oleh sesuatu perisian dan pangkalan data yang berkaitan dapat membezakan kejayaan antara syarikat dengan pesaingnya yang lain. Fitur kepintaran serta kelebihan fungsi yang ada pada perisian juga dapat membezakan antara dua produk yang sama. Jadi perisian merupakan faktor penting dalam bidang perkomputeran yang perlu dirancang serta dikaji supaya kebaikannya dapat digunakan sepenuhnya.

Sejarah juga menunjukkan perubahan yang berlaku kepada kepentingan perisian dari segi kos yang terbabit. Pada awal tahun 60-an, keutamaan yang pertama ialah bagi perolehan perkakasan dan penyenggaraan perkakasan. Pada waktu itu, kos perkakasan merupakan 80% dari kos keseluruhan perbelanjaan bagi peruntukan perkakasan dan perisian. Walau bagaimana pun, pada pertengahan tahun 60-an, kos perisian telah meningkat sehingga 30% dari kos keseluruhan (Mynatt 1990). Kini, kos perkakasan cuma mewakili sebanyak 20% dari jumlah kos perkakasan dan perisian (Boehm 1983). Disebabkan peningkatan kos perisian ini, kini keutamaannya ialah untuk menjadikan proses pembangunan perisian dijalankan secara berkesan.

Sebagai langkah untuk mengawal peningkatan kos perisian ini, faktor penyumbang kepada peningkatan ini perlu dikenal pasti dan masalahnya perlu diatasi. Terdapat beberapa faktor yang telah dikenal pasti menyumbang kepada peningkatan kos perisian. Salah satu faktor tersebut ialah peningkatan kos untuk kegiatan penyenggaraan. Mengikut satu laporan, lebih 60% daripada kos perisian adalah untuk penyenggaraan (Harrison 1987). Penyenggaraan perisian merupakan aktiviti kejuruteraan perisian yang dijalankan sebaik sahaja sistem perisian telah siap dibina dan dipasang.

Untuk menjadikan sesuatu perisian itu mudah untuk disenggarakan, perisian itu perlu mempunyai ciri-ciri seperti ralat yang minimum, dokumentasi yang lengkap, kod yang mudah dibaca, serta reka bentuk dan struktur perisian yang baik. Persoalannya sekarang bagaimanakah untuk menerapkan ciri-ciri tersebut semasa pembangunan perisian, supaya perisian yang terhasil menjadi mudah disenggara? Salah satu ciri tersebut ialah kod yang mudah dibaca. Seperti yang diketahui penyenggara perlu melibatkan dirinya dalam

pemahaman atur cara sebelum dapat membuat sebarang perubahan. Kefahaman atur cara pula bergantung kepada faktor kekompleksan atur cara yang ditulis. Sesuatu atur cara itu sukar untuk difahami jika darjah kekompleksannya tinggi.

Dengan penilaian yang berterusan, darjah kekompleksan atur cara ini mungkin boleh dikawal dan maklumat yang didapati boleh digunakan dalam penghasilan kod yang mudah dibaca. Terdapat banyak kaedah yang boleh digunakan untuk tujuan mengukur kekompleksan atur cara tetapi kebanyakan analisis hanya meliputi aspek kuantitatif tanpa mengambil kira aspek kualitatifnya. Aspek kuantitatif ini terhad kepada menghitung jumlah kendalian dan pengendali yang terkandung dalam atur cara. Aspek kualitatif pula memberi penekanan kepada aspek-aspek lain yang mempengaruhi pandangan pembina semasa membina atur cara itu. Justru itu, kertas ini seterusnya akan membincangkan aspek metrik perisian berkaitan dengan pengukuran kekompleksan. Seterusnya metodologi kajian akan diterangkan sesuai dengan keperluan kepada ukuran kualitatif yang banyak dipengaruhi oleh faktor tempatan pembina atur cara. Perbincangan seterusnya melibatkan analisis ke atas hasil kajian yang diperolehi. Pada akhir kertas ini, satu set indeks kekompleksan bahasa C akan dibentuk yang boleh digunakan dalam pengukuran metrik yang bersifat kualitatif.

METRIK PERISIAN

Kini pembangunan perisian telah dikenali sebagai kejuruteraan perisian dan untuk memantapkan kedudukannya sebagai satu cabang kejuruteraan, perisian itu perlu mampu untuk diukur dan dikuantitikan. Perkembangan ini telah banyak menimbulkan minat di kalangan penyelidik dalam aspek pengukuran perisian yang lebih dikenali sebagai metrik perisian. Perkataan metrik mempunyai kaitan dengan perkataan meter, iaitu satu alat atau mekanisma untuk mengukur dan merekodkan sesuatu jumlah. Manakala metrik pula adalah kuantiti yang didapati dari meter tersebut. Menurut Fenton dan Kaposi (1987), metrik perisian adalah ukuran dalam bentuk kuantiti yang dihasilkan secara algoritma atau mekanikal yang boleh didapati dari mana-mana sifat kitar hayat perisian. Umumnya, metrik membantu dalam memahami proses teknikal yang digunakan untuk membangunkan produk dan produk itu sendiri (Pressman 2001).

Metrik perisian dapat memberikan maklumat yang penting sepanjang kitar hayat perisian. Pfleeger dan Fitzgerald (1991) menyatakan ada tiga sebab utama mengapa perisian perlu diukur dan dikuantitikan. Pertama, ukuran

yang dihasilkan boleh digunakan untuk pencapaian produktiviti dan kualiti sesuatu perisian. Dengan pengukuran ini, perisian yang dihasilkan boleh dibandingkan, diberi nilai dan ditentukan kualitinya. Dengan adanya sasaran kualiti ini juga, perisian yang terbina dapat mengelakkan beberapa masalah biasa yang timbul semasa proses pembangunan perisian dijalankan; maka perisian yang terhasil adalah lebih mantap dan boleh dipercayai. Proses penghasilan sesuatu perisian juga dapat dipertingkatkan dan sumber yang diperuntukkan terutamanya sumber tenaga manusia dapat digunakan dengan lebih berkesan.

Kedua, ukuran yang dihasilkan semasa pembangunan perisian dijalankan dapat menunjukkan tempat-tempat pada fasa pembangunan yang perlu diberi lebih perhatian, analisis, atau pengujian. Pengukuran yang dihasilkan pada fasa reka bentuk, pengekodan, dan penyenggaraan dapat membaiki masalah yang timbul pada awal proses pembangunan dan ini tentunya dapat mengurangkan kos penyenggaraan. Perisian yang dibangunkan dapat dikawal dan diuruskan dengan lebih berkesan lagi kerana masalah yang dikesan pada setiap fasa telah diketahui dan diperbaiki. Pengukuran ini juga dapat memberi maklumat penting kepada pengurus projek dalam menjejaki dan mengawal perjalanan projek.

Ketiga, ukuran yang dihasilkan dapat membantu pihak pengurusan membuat perancangan dan jangkaan tentang projek masa akan datang berdasarkan maklumat yang dikumpulkan dari projek masa lalu. Sebagai contoh faedah hasil dari penggunaan peralatan serta metodologi yang baru dapat diperolehi dan dinilai. Justifikasi untuk memperolehi kelengkapan baru dan penyediaan latihan tambahan juga dapat dibuat dengan mudah. Kualiti produk yang dihasilkan serta keberkesanan proses yang menghasilkannya dapat dipertingkatkan untuk projek akan datang supaya sesuatu projek itu dapat dihasilkan dalam jangka masa dan belanjawan yang diperuntukkan.

Metrik kekompleksan perisian telah mula diperkenalkan pada awal tahun 70-an bertujuan untuk menganalisis kekompleksan atur cara dan keseluruhan perisian. Idea untuk menggunakan metrik kekompleksan adalah berasaskan kepada pengalaman iaitu kekompleksan atur cara menimbulkan masalah bukan sahaja pada fasa pembangunan tetapi juga pada fasa penyenggaraan perisian. Istilah kekompleksan perisian telah mula dibincangkan sejak tahun 70-an di kalangan penyelidik (Myers 1976; Yourdon & Constantine 1979 dan Fenton 1994). Myers (1976) pernah menulis tentang kekompleksan perisian yang berbunyi:

“Kekompleksan sukar untuk diberi satu takrifan yang tepat dan dikuantitikan. Walaubagaimanapun, bolehlah dikatakan yang kekompleksan sesuatu objek adalah pengukuran sebahagian dari usaha minda yang diperlukan dalam memahami objek tersebut.”

Seterusnya Myers (1976) memberi definisi iaitu kekompleksan sesuatu objek adalah satu fungsi bagi perhubungan antara komponen-komponen objek tersebut. Istilah kekompleksan sering juga dianggap sinonim dengan perkataan kefahaman atau penyenggaraan. Yourdon & Constantine (1979) juga berpendapat mengatakan masalah dalam pengaturcaraan berlaku kerana manusia sering membuat kesilapan dan mereka membuat kesilapan kerana kemampuan pemikiran mereka yang terbatas tentang kekompleksan. Oleh kerana manusia terlibat sepenuhnya dalam pembangunan sesuatu perisian dan reka bentuk sistem, istilah kekompleksan selalunya dilambangkan sebagai kekompleksan psikologi.

Kekompleksan psikologi merujuk kepada ciri-ciri yang menimbulkan kesukaran kepada manusia untuk memahami sesuatu perisian yang dibaca (Shneiderman 1980). Antara pengukuran kekompleksan yang terkenal yang telah dicipta pada pertengahan tahun 70-an ialah metrik saiz modul, metrik McCabe pada tahun 1976, metrik sains perisian Halstead pada tahun 1977 dan metrik McClure pada tahun 1978 (Halstead 1977; McCabe 1976 dan McClure 1978). Metrik ini sering digunakan kerana pengiraannya yang mudah dan penggunaannya yang umum untuk berbagai aplikasi serta penggunaannya yang meluas di kalangan industri (Martin 1983).

Shneiderman (1980) pula telah mengklasifikasikan metrik kekompleksan dalam tiga bentuk iaitu mantik, berstruktur, dan psikologi. Kekompleksan mantik melibatkan pengukuran dibuat terhadap ciri-ciri atur cara yang menimbulkan kesukaran dalam pembaikan. Kekompleksan berstruktur pula mengukur jumlah modul dan pautan antara modul tersebut dalam sesuatu atur cara. Manakala kekompleksan psikologi merujuk kepada ciri-ciri yang menimbulkan kesukaran kepada manusia dalam memahami perisian. Istilah kebolehfahaman sering digunakan untuk memberi makna kepada kekompleksan psikologi. Perbincangan yang lebih jelas dan terkini mengenai metrik kekompleksan ini boleh dilihat di dalam Zuse (1997) dan Zultner (1999).

Daripada perbincangan di atas, kelihatan ukuran nilai statik secara kuantitatif tentunya tidak mencukupi untuk menyatakan bahawa metrik yang diperolehi adalah tepat dan boleh dipercayai. Untuk itu satu model berasaskan unsur kualitatif telah dibentuk oleh Berns (1984) terhadap bahasa Fortran. Namun begitu, nilai kualitatif yang diberikan kepada unsur bahasa

pengaturcaraan dibuat berasaskan kepada pengalaman, intuisi dan pengetahuan beliau sahaja. Dan tentunya nilai pemberat ini tentulah tidak melambangkan budaya tempatan di negara ini. Dengan yang demikian wajar sekali kajian terhadap penentuan nilai kualitatif dibuat terhadap bahasa yang lebih popular masa kini iaitu bahasa C dengan mengambil kira aspek budaya pembangunan perisian tempatan.

METODOLOGI KAJIAN

Kajian imperikal yang dibentangkan dalam kertas ini dibuat secara soal selidik. Satu set soalan telah disediakan yang merangkumi perkara-perkara berikut:

- Latar belakang responden
- Mendapatkan faktor yang mempunyai hubung kait dengan nilai kekompleksan yang diberikan oleh responden
- Nilai pemberat yang mewakili kekompleksan setiap unsur

Dengan perolehan maklumat di atas, matlamat kajian untuk mendapatkan nilai pemberat yang mewakili kekompleksan bagi setiap unsur bahasa C akan dapat dicapai. Nilai ini yang akan membentuk indeks kekompleksan tentulah lebih tepat memandangkan hasil kajian ini dibuat menurut pandangan jurutera perisian secara kolektif dalam persekitaran tempatan.

Kawasan Kajian

Kawasan kajian yang telah dipilih ialah organisasi MIMOS, UKM, ITM Shah Alam dan syarikat Time Engineering. Pemilihan tempat-tempat tersebut adalah berdasarkan kepada kemudahan untuk mendapatkan responden yang mempunyai pengalaman dalam pengaturcaraan bahasa C. Di samping itu kegiatan penulisan atur cara C secara komersial di negara ini agak terhad.

Reka bentuk sampel

Pensampelan secara rawak ringkas digunakan dalam kajian ini bagi kawasan kajian yang telah dinyatakan. Cara mendapatkan responden ialah dengan memilih secara rawak mana-mana jurutera perisian yang ditemui di tempat tersebut. Responden yang dipilih juga terdiri dari mereka yang pernah menulis atur cara bahasa C untuk pembinaan sistem di tempat tersebut. Di

syarikat Time Engineering umpamanya, responden telah dipilih oleh ketua projek masing-masing.

Saiz Sampel

Salah satu kerumitan dalam kajian ini ialah untuk mendapatkan responden yang pernah menulis atur cara C untuk aplikasi sebenar. Di samping itu, kesibukan responden dan keutamaan yang rendah diberikan kepada pengisian soal-selidik ini menyebabkan saiz sampel yang lebih banyak sukar diperolehi. Akhirnya, kajian ini telah berjaya mendapatkan saiz sampel berjumlah 30. Namun begitu, bilangan 30 ini, dipercayai dan diharapkan dapat menggambarkan pendapat di kalangan jurutera perisian tentang kekompleksan unsur bahasa C.

Pengumpulan Data

Pengumpulan data dijalankan secara primer melalui sumber pertama. Aktiviti pengumpulan data secara primer ini dibahagikan kepada dua iaitu penyediaan borang soal-selidik dan menemuduga responden. Data diperolehi dengan menggunakan borang soal-selidik yang mengumpul maklumat terhadap dua perkara utama iaitu latar belakang responden dan kekompleksan setiap unsur bahasa C. Secara amnya, hampir 99% soal-selidik itu mengandungi soalan tertutup dan cuma satu soalan terbuka. Soalan terbuka tersebut adalah untuk mengetahui pendapat responden tentang kegiatan penyenggaraan.

Dalam kajian ini, menemuduga responden merupakan langkah paling penting. Tempoh temuduga dengan borang soal selidik yang peruntukkan ialah selama seminggu. Borang tersebut seterusnya disemak semula dan dijalankan proses pengkodan dan penganalisan komputer. Proses pengkodan merupakan langkah memindahkan data-data yang berbentuk kualitatif ke bentuk kuantitatif. Seterusnya memindahkan kod-kod itu ke dalam komputer untuk proses penganalisan yang dilakukan menggunakan fungsi statistik pakej Lotus 123. Jenis penganalisan yang dijalankan termasuklah analisis berperihal yang juga dikenali sebagai keterangan berstatistik iaitu taburan kekerapan dan analisis regresi linear.

ANALISIS DATA DAN PERBINCANGAN

Latar Belakang Responden

Butir-butir responden yang diambil kira ialah jumlah tahun pengalaman bekerja dan kelulusan akademik yang tertinggi. Secara keseluruhannya, berdasarkan Jadual 1 jelas menunjukkan bahawa bilangan responden dalam kumpulan jumlah tahun bekerja antara 1 hingga 5 tahun adalah paling tinggi iaitu sebanyak 66.7%. Manakala bilangan responden dalam kumpulan akademik ijazah adalah paling tinggi iaitu sebanyak 76.7%. Taburan pengalaman kerja dan tahap kelulusan akademik menjadi sedemikian rupa mungkin disebabkan oleh responden dalam peringkat ini banyak terlibat dalam pengaturcaraan bahasa C.

Jadual 1:Kekerapan dan Peratusan Pengalaman Kerja dan Akademik

PENGALAMAN BEKERJA	KEKERAPAN	PERATUS
Kurang dari 1 tahun	5	16.7
1 - 5 tahun	20	66.6
6 - 10 tahun	5	16.7
KELULUSAN AKADEMIK		
Ijazah	23	76.7
Sarjana	7	23.3

Jadual 2:Kekerapan dan Peratusan Pengalaman dan Tahap Kemahiran

PENGALAMAN PENGATURCARAAN	KEKERAPAN	PERATUS
Kurang dari 1 tahun	2	6.67
1 - 5 tahun	25	83.3
6 - 10 tahun	3	10
TAHAP KEMAHIRAN		
Kurang baik	3	10
Sederhana	13	43.3
Baik	11	36.7
Sangat Baik	3	10

Peratusan responden mengikut tahun pengalaman dalam pengaturcaraan bahasa C dan tahap kemahiran dipaparkan dalam Jadual 2. Didapati bahawa majoriti daripada responden mempunyai pengalaman pengaturcaraan antara 1 hingga 5 tahun atau 83.3% daripada keseluruhan responden. Manakala bilangan responden yang mempunyai tahap kemahiran

sederhana dan baik adalah tinggi iaitu sebanyak 43.3% dan 36.7% masing-masing.

Jadual 3: Kekerapan dan Peratusan Maklumat Pembangunan Perisian

JENIS PENGKOMPIL	KEKERAPAN	PERATUS
Borland C++	25	83.3
Turbo C++	23	76.7
Gnu C	4	13.3
Microsoft C	12	40
JENIS APLIKASI		
Pengurusan	6	20
CAI/CAL	9	30
Kejuruteraan	6	20
Komunikasi	13	43.3
PERSEKITARAN		
DOS	28	93.3
WINDOWS	17	56.7
UNIX	13	43.3
VMS	1	3.3
PIAWAIAN		
Ada	14	46.7
Tiada	16	53.3
METODOLOGI		
SSADM	9	30
JSDM	3	10
HIPO	4	13.3
OOMT	9	30

Butir-butir lain yang diambil kira termasuklah jenis pengkompil, jenis aplikasi yang telah dibangunkan, jenis persekitaran, metodologi serta piawaian yang telah digunakan dalam membangunkan perisian. Berdasarkan Jadual 1, jelas menunjukkan majoriti responden berpengalaman menggunakan jenis pengkompil Borland C++ dan Turbo C++ iaitu sebanyak 25 (83.3%) dan 23 (76.7%) masing-masing. Majoriti dari aplikasi yang telah dibangunkan oleh responden ini adalah berjenis komunikasi iaitu sebanyak 13 (43.3%). Tambahan pula, majoriti dari aplikasi ini telah dibangunkan menggunakan persekitaran Dos iaitu 28 (93.3%), diikuti oleh Windows iaitu 17 (56.6%). Sebanyak 53.3% responden tiada menggunakan piawaian sama ada yang telah ditetapkan oleh

organisasi atau bahasa itu sendiri. Majoriti daripada responden menggunakan metodologi SSADM dan OOMT iaitu sebanyak 9 (30%) semasa pembangunan perisian dijalankan.

Merujuk kepada Jadual 4, didapati majoriti daripada responden terlibat dengan kegiatan penyenggaraan iaitu seramai 27 orang (90%). Daripada yang terlibat, cuma seramai 5 orang (16.7%) berpendapat kegiatan ini telah diberi ganjaran atau insentif. Majoriti daripada responden iaitu seramai 16 orang (53.3%) yang terlibat dengan kegiatan penyenggaraan bukan merupakan pembangun yang asal. Didapati hanya 12 orang (40%) responden menggunakan alat bantuan dalam menjalankan kegiatan penyenggaraan dan majoriti dari responden ini menggunakan alat penyahpepijat iaitu sebanyak 10(83.3%).

Jadual 4: Kekerapan Dan Peratusan Kegiatan Penyenggaraan

PENYENGGARAAN	KEKERAPAN	PERATUS
Terlibat	27	90
Tidak Terlibat	3	10
GANJARAN/INSENTIF		
Ada	5	16.7
Tiada	25	83.3
PEMBANGUN ASAL		
Ya	14	46.7
Tidak	16	53.3
ALAT BANTUAN		
CASE	1	8.3
Alat Penyenggaraan	1	83.4
Alat Debugger	10	8.3

Jadual 5 : Kekerapan Dan Peratusan Teknik & Keperluan Pakar

PENGETAHUAN TEKNIK KEKOMPLEKSAN	KEKERAPAN	PERATUS %
Tiada	4	13.3
Sedikit	10	33.3
Sederhana	11	36.7
Amat Baik	5	16.7
TEKNIK YANG DIGUNAKAN		
Fungsi Titik	2	6.7
Teknik Saiz Atur cara	7	23.3
Teknik "Halstead Software Science"	1	3.3
KEPERLUAN PAKAR PENYENGGARAAN		
Tidak Diperlukan	7	23.3
Sentiasa Diperlukan	3	10
Bergantung Kepada Projek	17	56.7
Bergantung Kepada Peruntukan	13	10

Merujuk kepada Jadual 5, secara keseluruhannya didapati majoriti responden mempunyai tahap pengetahuan yang sedikit dan sederhana tentang teknik kekompleksan atur cara iaitu sebanyak 10 (33.3%) dan 11 (36.7%) masing-masing. Daripada responden yang tahu, hanya terdapat seramai 7 orang (23.3%) yang menggunakan teknik saiz atur cara untuk menentukan kekompleksan atur cara. Seramai 17 orang (56.7%) responden berpendapat keperluan khidmat pakar dalam menjalankan tugas penyenggaraan perisian adalah bergantung kepada projek yang dibangunkan.

Faktor Hubungan Kekompleksan

Analisis regresi linear telah dibuat untuk mencari faktor yang mungkin mempunyai hubungan dengan nilai kekompleksan yang telah diberikan oleh responden. Merujuk kepada Jadual 6, hasil analisis mendapati 9 faktor yang mempunyai hubungan kait berkadar R sebanyak 65%.

Jadual 6 : Hasil Analisis Regresi Linear

Regression Output:	
Constant	5.126552
Std Err Of Y Est	3.530689
R Squared	0.424163
No. of Observations	30
Degrees of Freedom	20
R	0.651277
	F1 F2 F3 F4 F5 F6 F7 F8 F9
X Coefficient(s)	-4.01 -3.33 -1.27 -1.19 -1.05 -0.08 0.03 2.04 3.58

Walaupun kadar R ini tidak begitu besar dengan tidak menghampiri nilai 100%, namun begitu hubungan wujud antara faktor dan nilai kekompleksan. Ini juga menyatakan terdapat lagi faktor-faktor lain yang mempunyai hubungan tetapi tidak didapati dari kajian ini. Sembilan faktor yang didapati hasil dari kajian ini diperihalkan dalam Jadual 7.

Pembentukan Indeks Kekompleksan

Setiap responden telah mengumpukkan nilai pemberat bagi mewakili kekompleksan setiap unsur bahasa C. Cuma terdapat 107 unsur yang telah disenaraikan untuk diberi nilai oleh setiap responden. Pemilihan unsur-unsur ini dibuat kerana unsur ini merupakan unsur-unsur yang asas dalam pengaturcaraan bahasa C. Nilai yang dibenarkan bagi setiap reponden ialah antara 0 hingga 20, iaitu 0 untuk terlampau mudah dan 20 untuk terlampau kompleks. Satu nilai purata telah dihasilkan untuk setiap unsur dengan menggunakan fungsi @AVG. Nilai inilah yang menjadi input kepada indeks kekompleksan seperti yang ditunjukkan oleh Jadual 8.

Jadual 7: Faktor Hubungan Kekompleksan

Bil.	Faktor	Penerangan
1.	Penggunaan teknik kekompleksan atur cara	Didapati responden yang berpengalaman dalam menggunakan teknik kekompleksan atur cara akan mengurangkan nilai kekompleksan yang diberikan iaitu dengan kadar sebanyak 4. Ini mungkin disebabkan dengan menggunakan teknik tersebut responden lebih berpengalaman dalam menentukan kekompleksan.
2.	Penggunaan alat bantuan dalam kegiatan penyenggaraan	Didapati dengan bertambahnya penggunaan alat bantuan, berkurangan nilai kekompleksan yang diberikan dengan kadar 3.33. Ini mungkin disebabkan oleh dengan adanya alat-alat ini dapat membantu responden dalam memahami unsur dengan lebih jelas dalam masa yang singkat.
3.	Pengalaman bekerja	Didapati responden yang bertambah pengalaman kerja berkurangan nilai kekompleksan yang diberikan dengan kadar sebanyak 1.27. Ini mungkin disebabkan majoriti dari responden yang mempunyai pengalaman bekerja ini juga mempunyai pengalaman yang serupa dalam pengaturcaraan bahasa C.
4	Pengalaman pengaturcaraan bahasa C	Didapati bertambah pengalaman responden dalam pengaturcaraan berkurangan nilai kekompleksan yang diberikan dengan kadar pengurangan sebanyak 1.19. Ini mungkin disebabkan oleh dengan bertambahnya pengalaman dalam pengaturcaraan bahasa C responden terdedah dengan berbagai gaya penulisan dan ini menjadikan mudah untuk memahami unsur-unsur bahasa C.
5.	Tahap kemahiran pengaturcaraan	Didapati bertambah mahir responden dalam bahasa C berkurangan nilai kekompleksan yang diberikan dengan kadar 1.05. Ini mungkin disebabkan kemahiran yang bertambah akan menjadikan responden cekap menggunakan unsur-unsur bahasa C.
6.	Piawaian yang diikuti	Didapati responden yang mengikuti piawaian pengaturcaraan bahasa C akan mengurangkan nilai kekompleksan yang diberi dengan kadar sebanyak 0.08. Ini mungkin disebabkan dengan mengikuti satu piawaian, unsur-unsur piawai yang jelas telah digunakan dan ini seterusnya menurunkan kadar kekompleksannya.
7.	Penglibatan dalam kegiatan penyenggaraan	Didapati responden yang terlibat dalam kegiatan penyenggaraan akan menambahkan nilai kekompleksan dengan kadar 0.03. Ini mungkin disebabkan apabila seseorang terlibat dengan kegiatan ini mereka terdedah dengan berbagai gaya penggunaan unsur dan faktor ini mungkin menyebabkan nilai ini bertambah.
8.	Metodologi	Didapati responden yang berpengalaman dalam menggunakan banyak metodologi akan menambahkan nilai kekompleksan yang diberikan dengan kadar 2.04. Ini mungkin disebabkan apabila seseorang itu berpengalaman dalam berbagai metodologi, bertambah pengalaman dalam penulisan atur cara. Dan ini mungkin akan meningkatkan nilai yang diberikan kepada setiap unsur.
9.	Pengetahuan kekompleksan atur cara	Didapati bertambah pengetahuan responden tentang kekompleksan atur cara, bertambah nilai kekompleksan yang diberikan dengan kadar 3.58. Ini mungkin disebabkan faktor pengetahuan akan menambahkan kepekaan tentang kekompleksan dan seterusnya memberi kesan dalam meningkatkan nilai yang diberikan.

KESIMPULAN

Penentuan kekompleksan atur cara amat penting terutamanya dalam proses penganggaran sumber bagi pembangunan dan poenyenggaraan perisian. Kaedah analisis statik kod sumber merupakan kaedah penentuan kekompleksan yang popular. Bagaimana pun kaedah yang banyak kedapatan pada masa ini hanya melibatkan aspek kuantitatif. Terdapat saranan bahawa kekompleksan juga seharusnya melibatkan penilaian aspek kualitatif. Untuk kertas ini telah berjaya memaparkan hasil kajian yang menunjukkan bagaimana nilai bagi aspek kualitatif boleh dibentuk yang banyak dipengaruhi oleh faktor persekitaran dan budaya kerja.

Jadual 8: Indeks Kekompleksan Unsur Bahasa C

Nama Simbolik	Ind	Unsur Pengendali	Ind	Fungsi Lazim	Ind	Unsur Arahan	Ind
Array	4.7	+, -, *, =, !=, !	2.7	isupper	4.6	auto	5.2
Variable	3.2	%, /, >, <, & &	2.8	strcat, strcmp	4.4	register	7.2
Constant	2.6	&, , ^, =, +=, -=, *=	3.5	strlen	3.9	global	4.2
Label	3.7	<<, >>	3.9	fgetc, ungetc	5.2	void	4.3
Pointer	8.0	++, --, >=, <=	3.2	h			
Structure	6.6	->, ?:	4.0	Unsur Arahan	Ind	break	2.9
Argument	5.2	.	3.6	static	4.9	continue	3.7
Parameter	4.5	~	3.3	if	2.5	default	2.8
File	5.0	<<=, >>=	4.7	else	2.7	while	3.0
Enum	4.8	/=, %=	4.1	else if	2.8	do	3.3
Union	6.3	&=, ^=, /=	4.5	switch	3.0	for	3.2
Unsur Pemalar	Ind	Fungsi Lazim	Ind	case	2.9	assignment	3.5
int	2.2	fopen, fclose	4.9	goto	3.8	null statement	3.7
long	2.5	fread, fwrite	4.9	return	3.2	}	2.7
unsigned	2.4	fputc	5.0	casting	4.6	typedef	3.9
signed	2.5	fscanf	5.1	{	2.7	definition statement	3.9
short	2.6	read	3.8	function	4.2	ref to built-in function	5.4
float	3.2	kbhit, scanf	4.7	prototype			
double	3.5	getc, printf, putchar	3.6	module name	3.5	#define	3.2
octal	4.3	exit	3.3	function	5.7	#if	5.2
hexadecimal	3.4	write, isdigit, islower	4.2	return value			
char	2.8	isprint	4.8	function	4.3	#ifdef	5.6
		strcpy	4.0	#endif	5.2	#ifndef	6.0
				#else	5.4	#include	3.3
				dereferencing	8.5	extern	4.2
				change value	5.1		

Terdapat 9 faktor yang mempunyai hubungan dengan nilai kekompleksan yang diberikan seperti yang diperihalkan dalam Jadual 7. Daripada jumlah ini, enam adalah perkaitan negatif iaitu pengalaman bekerja, tahun pengalaman dalam pengaturcaraan C, tahap kemahiran pengaturcaraan, mengikut piawaian semasa pengaturcaraan, alat bantuan penyenggaraan, dan penggunaan teknik kekompleksan atur cara. Manakala tiga perkaitan positif ditemui antara metodologi, penglibatan dalam kegiatan penyenggaraan, dan pengetahuan tentang teknik dengan nilai kekompleksan. Secara keseluruhannya perkaitan ini cuma mempunyai hubungan sebanyak 65% namun ianya tetap menunjukkan faktor-faktor ini mempunyai hubungan. Kajian ini seterusnya telah berjaya menghasilkan satu set indeks kekompleksan unsur bahasa C yang mewakili aspek kualitatif berasaskan budaya setempat. Hasil kajian ini telah pun digunakan untuk membina alat penganalisis kekompleksan seperti yang dibincangkan oleh Norliza (1998).

Selain daripada sembilan faktor yang dibincangkan, masih terdapat faktor lain yang mempunyai kaitan dengan nilai kekompleksan yang masih belum ditemui melalui kajian ini. Oleh itu kajian lanjut boleh dilakukan untuk menambah senarai faktor kualitatif yang mungkin mempengaruhi kekompleksan pengguna bahasa atur cara C. Pembangunan indeks kekompleksan menggunakan pendekatan yang sama boleh juga ditumpukan kepada bahasa pengaturcaraan yang lebih moden seperti Java dan C++. Kajian dalam bahasa-bahasa ini mungkin lebih mencabar memandangkan kedua-duanya bercirikan bahasa visual yang sifatnya sangat kualitatif.

RUJUKAN

- Berns, G. M. 1984. Accessing software maintainability. *Communications of the ACM*. January, 27 (1): 14-23.
- Boehm, B. W. 1983. The hardware/software cost ratio: Is it a myth?. *IEEE Computer*. 16(3).
- Fenton, N. E. & A. A. Kaposi, 1987. Metrics and software structure. *Journal of Information and Software Technology*. 29(7), 301-320.
- Fenton, N. E. 1994. "Software Measurement: A Necessary Scientific Basis," *IEEE Trans. Software Engineering*, Vol. SE-20, No. 3, pp 199-206.
- Halstead, M. H. 1977. *Elements of Software Science*. New York, Elsevier Science Publishing Co.
- Harrison, R. 1987, "Maintenance Giant Sleeps Undisturbed in Federal Data Centers," *Computerworld* (March 9, 1987): 81-86.
- McCabe, T. J. 1976. A complexity measure. *IEEE Transaction on Software Engineering*, 2(4), 308-320.

- McClure, C. 1978. *Reducing COBOL Complexity Through Structured Programming*. New York, van Nostrand Reinhold Co., 77-121.
- Myers, G. J. 1976. *Software Reliability – Principles and Practices*. New York, John Wiley & Sons Inc.
- Mynatt, B. T. 1990. *Software Engineering With Student Project Guidance*. New Jersey, Prentice Hall Inc.
- Norlidza M. Y. dan A. B. Deraman, 1998, "PROCOMP: Alat Penganalisis Kekompleksan Atur cara Berasaskan Kaedah Kualitatif Dan Kuantitatif", *Sains Malaysiana Journal (Information Technology)*, Vol 28 (1&2), June 1998. 151-164.
- Pfleeger, S. L. & J. C. Fitzgerald, 1991. Software metrics toolkit: Support for selection, collection and analysis. *Information and Software Technology*. September, 427-482.
- Pressman, R. S. 2001. *Software Engineering: A Practitioners' Approach*. 5th edition. New York, McGraw Hill Book Co. Inc.
- Shneiderman, B. 1980. *Software psychology: Human factors in Computer and Information Systems*, Mass., Winthrop Publishers.
- Yourdan, E & L. Constantine, 1979. *Structured Design*, NJ., Prentice Hall Inc.
- Zultner, R.E. 1999. "What Do Our Metrics Mean?" *Cutter IT Journal*, Vol 2, No. 4, 11-19.
- Zuse, H., 1997. *A Framework of Software Measurement*, DeGruyter.