

Pengalaman Menggunakan Perisian Alice dalam Memperkenalkan Pengaturcaraan Berorientasikan Objek

Nursyahidah Alias
Nor Zalina Ismail
Mohd Rizal Razak

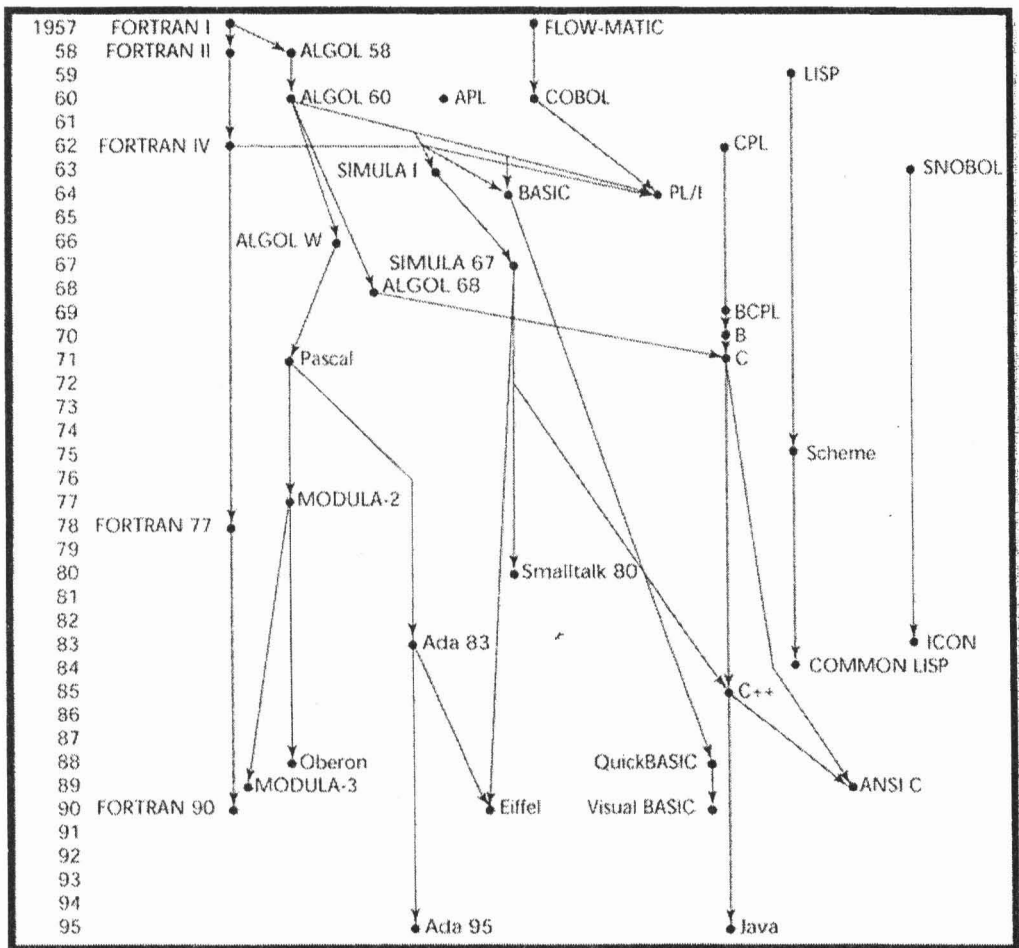
ABSTRAK

Pengaturcaraan berorientasikan objek adalah subjek yang perlu diajar kepada pelajar-pelajar pengaturcaraan. Kebiasaannya pelajar pada semester 1 dan 2 akan diperkenalkan dengan pengaturcaraan berstruktur terlebih dahulu. Pengaturcaraan berstruktur memfokuskan kepada penyelesaian masalah menggunakan rekabentuk prosedur logik yang melibatkan mengambil data input, memproses data input tersebut dan menghasilkan data output. Kemudian, pada semester 3 pelajar diperkenalkan pula dengan pengaturcaraan berorientasikan objek. Pengaturcaraan berorientasikan objek memfokuskan kepada penyelesaian masalah menggunakan rekabentuk objek dan data. Lazimnya pelajar mengalami masalah dalam memahami peralihan rekabentuk pengaturcaraan dari berstruktur kepada berorientasikan objek. Pensyarah juga mengalami masalah dalam menerangkan secara jelas perbezaan cara merekabentuk penyelesaian masalah tersebut. Perisian Alice dilihat dapat membantu pensyarah untuk digunakan sebagai alat bantu mengajar dalam memperkenalkan pengaturcaraan berorientasikan objek kepada pelajar. Perisian tersebut menggunakan objek Alice yang boleh dilihat secara fizikal bersama dengan tindakan yang boleh dibuat oleh Alice. Kertas ini membincangkan pengalaman menggunakan perisian Alice untuk digunakan sebagai alat bantu mengajar dalam pengajaran berorientasikan objek. Dengan menggunakan perisian tersebut objek Alice tersebut dapat difahami dengan jelas. Namun begitu, kebiasaannya apabila dalam pengajaran pengaturcaraan berorientasikan objek, objek dan data yang diwakilkan tidak dapat dilihat secara fizikal seperti Alice.

Katakunci: *Pengaturcaraan, Berorientasikan Objek, Alice*

Pengenalan

Aturcara komputer bukan hanya untuk memberitahu komputer apa yang perlu dibuat. Aturcara komputer adalah cara untuk memberitahu orang lain apa yang kita mahu komputer lakukan. Aturcara komputer yang elok adalah jika orang lain dengan mudah dapat memahami dan menghargai apa yang ingin dilakukan oleh pengaturcaraan (Dan et. al, 2009). Dan et.al juga mengatakan mempelajari aturcara komputer adalah mempelajari bagaimana berfikir untuk menyusun urutan arahan bagi melaksanakan tugas yang menjadi sebahagian perkara yang paling bernilai dalam mempelajari aturcara. Aturcara komputer dibangunkan dengan menggunakan bahasa pengaturcaraan. Pada masa kini terdapat pelbagai jenis bahasa pengaturcaraan antaranya adalah C (King, 2008), Java (Malik, 2010), PHP (Power, 2010) dan semakin banyak dari masa ke semasa. Rajah 1 menunjukkan sejarah perkembangan berdasarkan masa bagi bahasa pengaturcaraan (Department of Electronics Engineering, 2010).



Rajah 1 Sejarah perkembangan berdasarkan masa bagi bahasa pengaturcaraan

Menurut Jim (2010) terdapat tiga kategori bahasa pengaturcaraan iaitu bahasa pengaturcaraan berstruktur, pengaturcaraan berorientasikan objek dan bahasa yang melibatkan paradigma yang berbeza. Penerangan tentang ketiga-tiga bahasa pengaturcaraan tersebut adalah seperti berikut:

- Bahasa pengaturcaraan berstruktur adalah set idea dan susunan pemikiran yang jelas untuk membangunkan aturcara (Dijkstra, 1970).
- Bahasa pengaturcaraan berorientasikan objek adalah sejenis cara baru dalam menyelesaikan masalah dengan berasaskan objek dan membahagikan masalah kepada bahagian yang kecil (Ten Dyke, 1989).
- Bahasa yang melibatkan paradigma yang berbeza adalah teknik daripada pengaturcaraan berstruktur yang berasaskan kepada rekabentuk atas-bawah atau perincian langkah demi langkah (Floyd, 1978).

Kategori bahasa diatas menggambarkan rekabentuk bagi setiap bahasa tersebut.

Suh (2002) menyatakan terdapat lima jenis bahasa pengaturcaraan dan Graham (2011) pula menyatakan ia sebagai stail bahasa diungkapkan iaitu:

- Berprosedur
Pengaturcaraan berprosedur melaksanakan urutan pernyataan yang akan menghasilkan keputusan yang dikehendaki.
- Berfungsi
Pengaturcaraan berfungsi memfokuskan kepada nilai yang dipulangkan oleh fungsi, kesan yang dihasilkan apabila fungsi dilaksanakan dan keadaan yang berlaku selepas fungsi dilaksanakan.

- Berorientasikan objek
Pengaturcaraan berorientasikan objek bermatlamatkan kepada masalah yang dibahagikan kepada koleksi objek yang membolehkan masalah diselesaikan secara khusus.
- Skrip
Pengaturcaraan skrip lazimnya adalah berprosedur dan mengandungi elemen bahasa berorientasikan objek, tetapi pengaturcaraan skrip tidak menyokong pembangunan sistem yang besar.
- Logik
Pengaturcaraan logik membolehkan pernyataan diungkapkan dalam logik paras pertama dan membenarkan komputer menaakul tentang pernyataan tersebut.

Graham (2011) mengelaskan bahasa pengaturcaraan kepada tiga paras iaitu:

- Bahasa pengaturcaraan paras rendah
Bahasa ini memerlukan penulisan bahasa yang lebih spesifik kepada perincian yang berkaitan dengan set arahan komputer. Contohnya adalah seperti bahasa pengaturcaraan C.
- Bahasa pengaturcaraan paras tinggi
Bahasa ini biasanya boleh dikompil. Bahasa ini menggunakan simpanan memori secara terus. Contohnya adalah seperti Java dan C Sharp.
- Bahasa pengaturcaraan paras paling tinggi
Bahasa ini kebiasaannya menggunakan mesin maya dan kod perantara. Contohnya adalah seperti bahasa pengaturcaraan skrip dan SETL.

Setiap bahasa pengaturcaraan mempunyai sintak dan semantik tersendiri. Sintak adalah tatabahasa formal bagi bahasa pengaturcaraan dari segi bentuk dan struktur (Backhouse, 1979). Manakala semantik menunjukkan apa yang aturcara lakukan dari segi makna dan pelaksanaan keseluruhan (Hebler, 2009).

Pada masa kini, lazimnya terdapat perisian yang mempunyai antara muka yang membantu memudahkan penulisan bahasa pengaturcaraan. Contohnya adalah seperti perisian Borland C++ (Miano et. al, 1997) atau TurboC++ (Garfield, 2012) digunakan untuk penulisan bahasa C++. Manakala BlueJ (Barnes & Kolling, 2012) atau JCreator (Xinox Software, 2010) digunakan untuk penulisan bahasa Java. Setiap perisian mempunyai stail antaramuka masing-masing (Suh, 2002). Perisian tersebut juga didatangkan bersama pakej (Suh, 2002) yang terdiri dari aturcara komputer yang telah diisytiharkan terlebih dahulu untuk kegunaan pengaturcara semasa menulis. Setiap bahasa ditulis berdasarkan kepada algoritma penyelesaian masalah melibatkan sintak, semantik dan model memori (Graham, 2011) masing-masing yang dipanggil aturcara komputer. Algoritma penyelesaian masalah melibatkan strategi penyelesaian yang optimum (Cook & Ibrahim, 2005) dan pengiraan komputer yang umum (Suh, 2002). Aturcara tersebut akan dikompil menggunakan pengkompil masing-masing untuk menyemak kesilapan sintak dalam bahasa masing-masing. Setiap pengkompil mempunyai kadar kekompleksan (Suh, 2002) yang tertentu yang perlu difahami oleh pengaturcara.

Masalah Pelajar

Silibus bagi pelajar Sains Komputer, UiTM Pahang lazimnya memerlukan pelajar untuk memulai mempelajari pengaturcaraan berstruktur diikuti dengan pengaturcaraan berorientasikan objek. Tujuan untuk mempelajari kedua-dua pengaturcaraan tersebut adalah untuk pelajar mengetahui dua rekabentuk penyelesaian yang berbeza bagi masalah yang sama. Namun, kedua-dua rekabentuk tersebut agak sukar untuk dizahirkan semasa sesi pengajaran dan pembelajaran. Ini kerana semasa proses pembelajaran tersebut kedua-dua rekabentuk diajar menggunakan bahasa pengaturcaraan sebagai alat bantu mengajar. Rekabentuk pengaturcaraan berstruktur diajar menggunakan bahasa C++. Manakala rekabentuk pengaturcaraan berorientasikan diajar menggunakan bahasa Java.

Semasa dalam proses pembelajaran konsep rekabentuk penyelesaian masalah pengaturcaraan, pelajar perlu juga memahami banyak perkara sampingan yang perlu dipelajari bersama seperti sintak, semantik, model memori, perisian yang digunakan, antaramuka perisian, pengkompil, pakej, algoritma dan pengiraan komputer

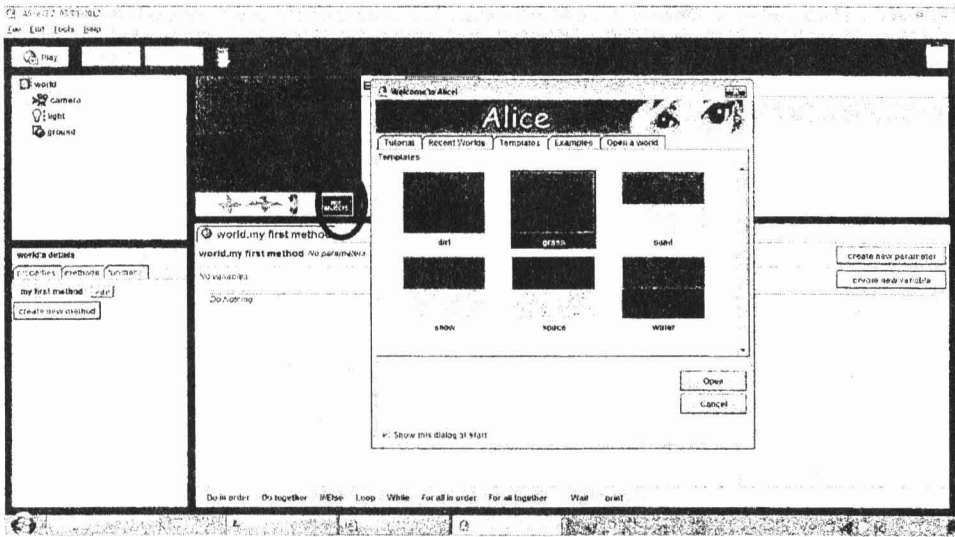
umum. Oleh yang demikian matlamat utama untuk mempelajari konsep rekabentuk penyelesaian masalah pengaturcaraan agak sukar difahami oleh pelajar.

Oleh yang demikian, sebagai tenaga pengajar dalam pengaturcaraan usaha untuk mendapatkan alat bantu mengajar yang memfokuskan kepada memahami konsep rekabentuk penyelesaian masalah pengaturcaraan dilakukan. Satu perisian yang dinamakan Alice cuba digunakan oleh tenaga pengajar untuk menentukan samada perisian tersebut sesuai digunakan sebagai alat bantu mengajar kepada pelajar Diploma Sains Komputer dalam memahami konsep rekabentuk penyelesaian masalah pengaturcaraan berorientasikan objek.

Perisian Alice

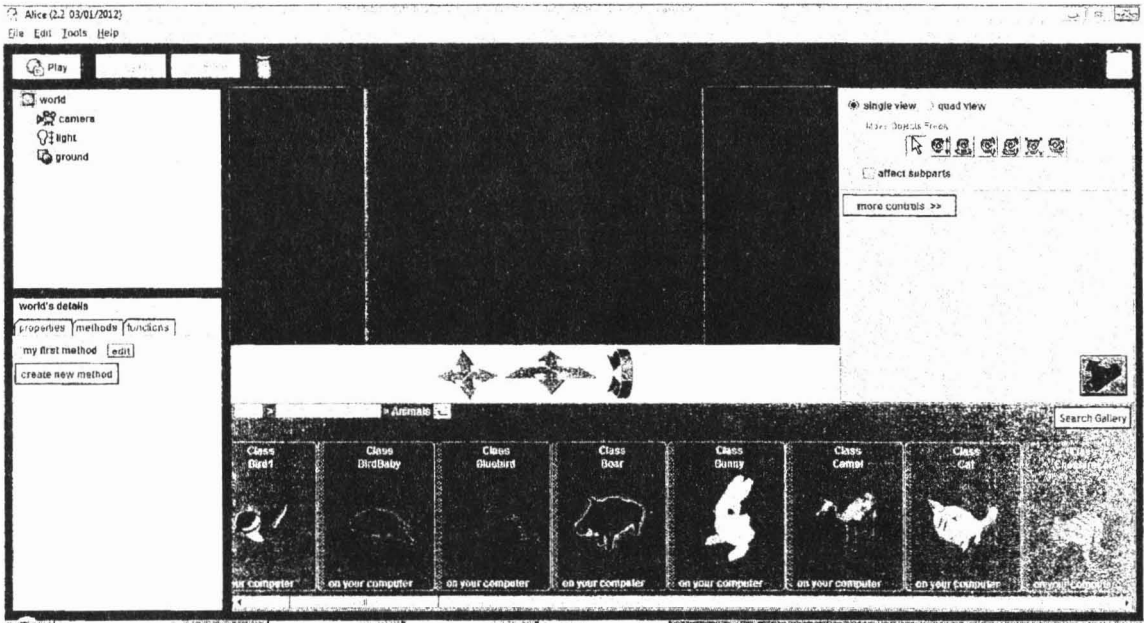
Perisian Alice telah dipelajari menggunakan buku (Dann et. al, 2009) dan perisian yang telah diinstall. Perisian Alice boleh dimuat turun di laman web rasminya (Carnegie Mellon University, 2012). Perisian tersebut mempunyai pakej yang telah sedia ada dengan pelbagai kelas. Setiap kelas telah mempunyai metodenya yang tersendiri.

Perisian Alice dimulakan dengan memilih rekabentuk Dunia (*world*) Alice seperti dalam Rajah 2. Dunia Alice juga adalah sejenis kelas.



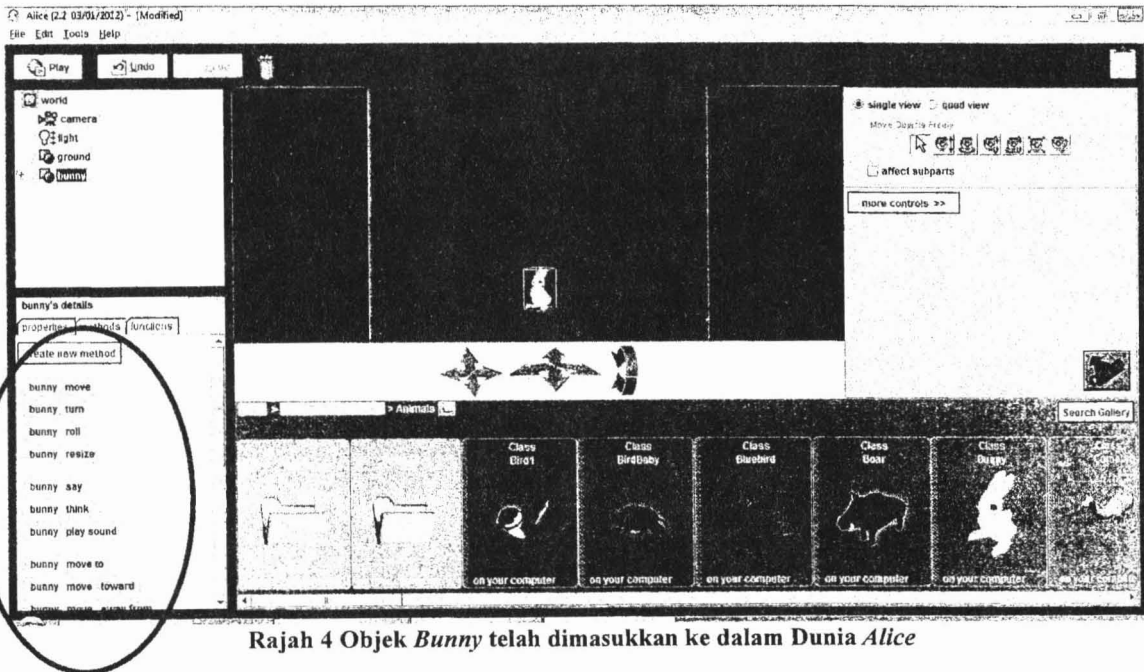
Rajah 2 Dunia Alice

Setelah Dunia dipilih objek boleh dimasukkan ke dalam Dunia Alice dengan mengklik kepada butang *ADD OBJECTS* seperti yang dibulatkan dengan tanda merah dalam Rajah 1. Objek boleh dihasilkan dengan hanya mengklik kepada galeri yang dikehendaki yang telah disediakan oleh Alice. Dalam perisian Alice, objek diwakilkan sebagai benda yang boleh dilihat secara jelas. Setiap *Gallery* mempunyai kelas masing-masing seperti yang ditunjukkan dalam Rajah 3.



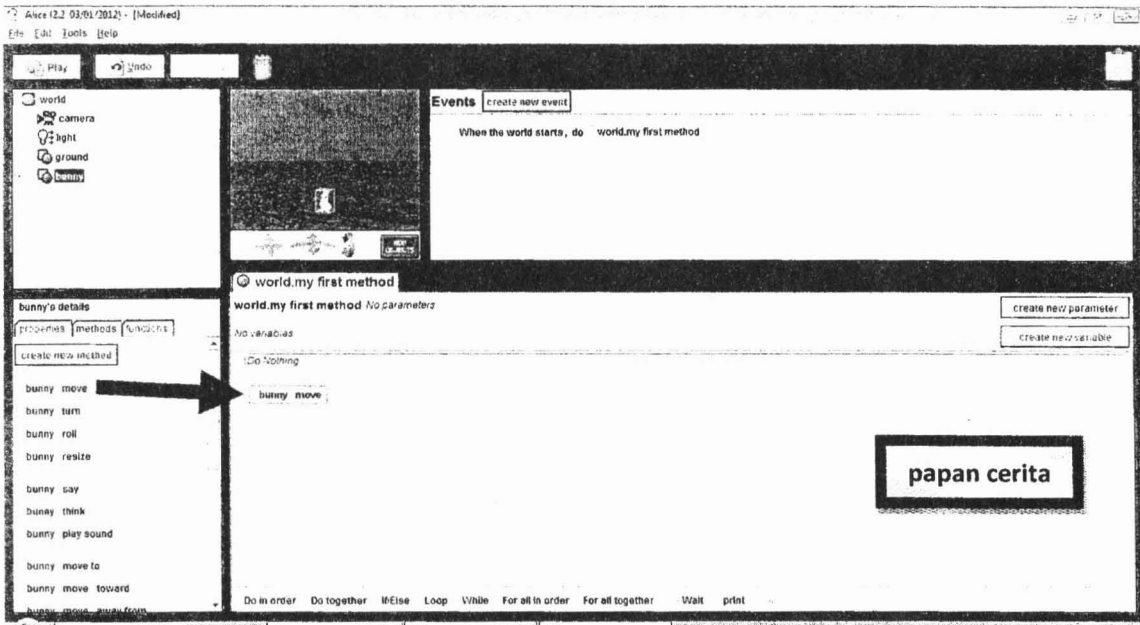
Rajah 3 Kelas yang terdapat dalam *Gallery Animals*

Rajah 4 pula menunjukkan objek *Bunny* telah dipilih untuk dimasukkan ke dalam Dunia Alice dengan hanya mengklik kepada ikon *Class Bunny*. Kelas *Bunny* mempunyai metod yang tersendiri seperti yang dibulatkan dengan tadnda merah dalam Rajah 4.



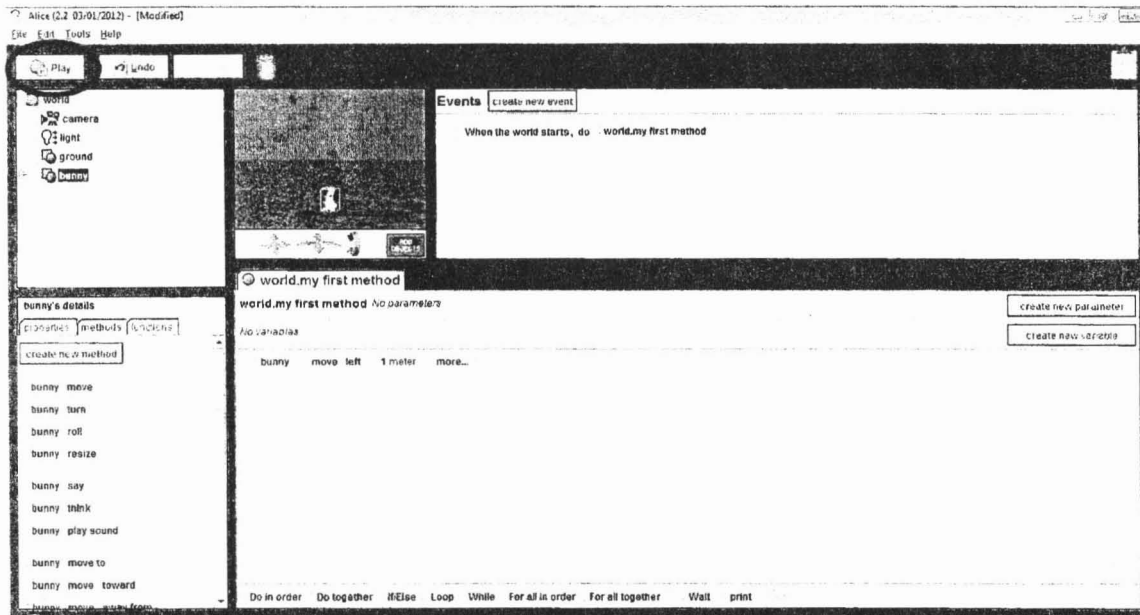
Rajah 4 Objek *Bunny* telah dimasukkan ke dalam Dunia Alice

Setelah selesai memilih objek, algoritma pelaksanaan aturcara boleh dilakukan *drag and drop* metod yang akan dilaksanakan oleh Bunny ke dalam papan cerita seperti yang ditunjukkan oleh Rajah 5.



Rajah 5 Melaksanakan *drag and drop* metod yang akan dilaksanakan oleh Bunny ke dalam papan cerita

Papan cerita adalah tempat untuk menyusun aturcara yang perlu dilaksanakan oleh *Bunny*. Setelah itu, pelaksanaan metod tersebut boleh diuji dengan dengan mengklik butang *Play* seperti yang dibulatkan dalam Rajah 6.



Rajah 6 Papan cerita yang mempunyai aturcara yang perlu dilaksanakan oleh Bunny dan butang *Play*

Setelah aturcara dilaksanakan Bunny akan bergerak ke kiri. Terdapat pelbagai contoh dan latihan yang disediakan dalam Alice yang boleh dicuba oleh sendiri pengguna dengan berpandukan kepada buku Dan et. al (2009) . Penerangan di atas telah merangkumi konsep objek, kelas, metod dan menguji aturcara dalam pengaturcaraan rekabentuk berorientasikan objek yang kadangkala memerlukan masa dua jam untuk menerangkan kepada pelajar konsep tersebut jika menggunakan bahasa pengaturcaraan.

Penutup

Perisian Alice amat memfokuskan konsep pengaturcaraan berorientasikan objek tanpa terikat kepada mana-mana bahasa pengaturcaraan. Ini kerana bahasa pengaturcaraan mempunyai sintak dan semantik yang perlu difahami terlebih dahulu sebelum menggunakannya. Pemahaman terhadap konsep pengaturcaraan berorientasikan objek lebih utama untuk dipelajari terlebih dahulu. Setelah memahami konsep tersebut pemahaman terhadap bahasa pengaturcaraan akan bertambah mudah kerana bahasa tersebut melaksanakan konsep yang sama. Namun begitu, silibus Diploma Sains Komputer memerlukan pelajar mempelajari dan menguasai bahasa pengaturcaraan bersama dengan konsep pengaturcaraan disebabkan kekangan masa 14 minggu pengkuliah yang diberikan. Oleh yang demikian, sebagai cadangan penyelesaian pada masa akan datang, bengkel penggunaan perisian Alice boleh dikendalikan sebelum subjek pengaturcaraan berorientasikan objek dimulakan diluar waktu pengkuliah.

Rujukan

- Backhouse, R.C. (1979). *Syntax of Programming Languages: Theory and Practice*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA.
- Barnes, D.J & Kolling, M. (2012). *Objects First with Java - A Practical Introduction using BlueJ* Fifth edition. Prentice Hall / Pearson Education.
- Carnegie Mellon University. (2012). *Alice: An educational Software that teaches students computer programming in 3D environments*. www.alice.org
- Cook, W.R. & Ibrahim, A.H. (2005). *Integrating Programming Languages & Databases: What's the Problem?*
- Dann, W.P., Cooper, S. & Pausch, R. (2009). *Learning to Program with Alice*, Second Edition. New Jersey. Pearson Education.
- Department of Electronics Engineering. (2010). *Evolution and History of Programming Languages Software/Hardware/System*. Nota.
- Dijkstra, E. W. (1970). *Structured programming*. Nato Software Engineering Conference 1969. Itali
- Floyd, R.W. (1978). *The Paradigms of Programming*. ACM Turing Award Lecture. *Communication of ACM*, Vol 22 (8).
- Garfield, N.F. (2012). *Turbo C++*. Anim Publishing.
- Graham, P. (2011). *Programming Languages Usage and Design Problems*. [Softpanorama.org](http://softpanorama.org). www.softpanorama.org
- Hebler, J., Fisher, W., Perez-Lopez, A. & Blace, R. 2009. *Semantic Web Programming*. Wiley Publishing, Inc. Indianapolis.
- Jim Hurst. (2010). *Types of Programming Languages*. Handbook.
- King, K. N. (2008). *C Programming: A Modern Approach*, 2nd Edition. Amazon.com
- Malik, D. S. (2010). *Java Programming: From Problem Analysis to Program Design*: 4th (fourth) Edition. Cengage Learning.
- Miano, J., Cabanski, T. & Howe, H. (1997). *Borland C++ Builder: the Definitive C++ Builder Problem Solver*. Waite Group Press.
- Power, D. (2010). *PHP Solutions: Dynamic Web Design Made Easy*, 2 edition. friendsofED.
- Suh, E. (2002). *The Tower of Babel – A Comparison Programming Languages*. Code Journal. <http://www.cprogramming.com/langs.html>

Ten Dyke, R. P. (1989). Object-Oriented Programming. IBM System Journals, Vol. 12 (3): 465 - 478. New York, USA.

Xinox Software. (2010). JCreator. The Netherlands. <http://www.jcreator.com/index.htm>

NURSYAHIDAH ALIAS, NOR ZALINA ISMAIL, MOHD RIZAL RAZAK, Universiti Teknologi MARA Pahang, syahidah@pahang.uitm.edu.my, nza1601@pahang.uitm.edu.my, dragon_admire007@pahang.uitm.edu.my