

CODEROOM: SISTEM PEMBELAJARAN PENGATURCARAAN KOMPETITIF DENGAN PENILAIAN AUTOMATIK

Mohamad Hakim bin Mohamad Zulkufli
Marini Abu Bakar

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia

ABSTRAK

Teknik pengajaran pengaturcaraan secara tradisional tidak lagi mampu untuk memberikan kesan baik yang efektif kepada kemahiran pengaturcaraan pelajar oleh kerana teknik tersebut tidak memberikan semangat atau mendorong pelajar untuk mengaturlara dengan lebih baik. Kajian telah menunjukkan bahawa pelajar akan belajar dengan lebih baik sekiranya mereka didorong oleh sesuatu motivasi. Sebuah persekitaran pengajaran pengaturcaraan yang kompetitif perlu diwujudkan untuk memberikan dorongan dan motivasi tersebut kepada pelajar. Penggunaan PC² di Fakulti Teknologi dan Sains Maklumat (FTSM) di Universiti Kebangsaan Malaysia (UKM) telah menimbulkan beberapa masalah seperti tidak boleh dicapai di luar waktu makmal mahupun luar daripada FTSM, pemarkahan secara nombor binar dan juga ketiadaan bank soalan. Akibat daripada sifat PC² yang hanya untuk digunakan dipertandingan pengaturcaraan, ianya tidak boleh digunakan untuk latihan pengaturcaraan secara berterusan. CodeRoom akan dibangunkan untuk mewujudkan dan mengintegrasikan persekitaran pembelajaran pengaturcaraan yang kompetitif. Sebuah sistem pemarkahan dan penilaian automatik juga akan dibangunkan untuk tujuan pembelajaran berterusan. Pembangunan CodeRoom akan dibuat mengikut metodologi *eXtreme Programming* (XP). Hasilnya dijangka akan dapat memenuhi segala keperluan pelajar untuk meningkatkan kemahiran pengaturcaraan mereka selain daripada mewujudkan persekitaran yang kompetitif untuk mendorong mereka agar terus maju. Di samping itu, hasil yang dijangka juga akan memudahkan pensyarah dengan meringankan beban mereka menggunakan sistem penilaian automatik dan lain-lain.

1 PENGENALAN

Pengaturcaraan komputer merupakan salah satu kursus yang penting bagi pelajar Sains Komputer dan Teknologi Maklumat. Pada tahun 2015, bilangan enrolmen pelajar warganegara Malaysia ke Institusi Pengajian Tinggi Awam (IPTA) serta pelajar dalam bidang “*Science, Technology, Engineering and Mathematics*” (STEM) seramai 514,233 orang. Daripada jumlah itu, 78,329 daripadanya merupakan pelajar yang telah mendaftar untuk bidang Sains, Matematik dan Komputer. Indikator ini menunjukkan bahawa 15% daripada jumlah enrolmen pelajar IPTA merupakan pelajar yang mempelajari pengaturcaraan komputer sebagai satu daripada subjek asas bagi kursus masing-masing.

Enrolmen pelajar dalam bidang Kejuruteraan, Pembuatan dan Pembinaan pada tahun 2015 pula adalah 118,166 orang iaitu hampir 23% daripada jumlah keseluruhan enrolmen pelajar warganegara di IPTA. (Kementerian Pendidikan Tinggi Malaysia 2016.) Pelajar

Kejuruteraan, Pembuatan dan Pembinaan juga mempelajari subjek pengaturcaraan komputer sebagai satu kemahiran asas untuk bidang mereka. Hal ini selaras dengan kenyataan bahawa pengaturcaraan merupakan salah satu kemahiran asas yang amat diperlukan dalam kelompok bidang Sains, Teknologi, Kejuruteraan dan Matematik atau lebih dikenali sebagai STEM. Untuk menerajui dalam mana-mana bidang STEM seseorang itu perlu mahir dalam pengaturcaraan komputer. Daripada statistik tersebut iaitu 38% pelajar STEM, dapat dilihat bahawa jumlah pelajar yang sangat ramai perlu mengambil kursus pengaturcaraan komputer.

Banyak penyelidikan dalam dan luar negara mendapati bahawa pelajar di universiti sering mempunyai masalah untuk memahami dan mahir dalam pengaturcaraan komputer. Kajian yang dilakukan oleh Lister et al. (2004) mendapati bahawa pelajar tidak mempunyai ilmu pengetahuan dan kemahiran yang diperlukan untuk menyelesaikan masalah. Ini menyebabkan pelajar sukar menguasai kemahiran pengaturcaraan. Akibatnya yang ketara dapat dilihat ketika peperiksaan atau ujian makmal pengaturcaraan dilakukan. Kebanyakan pelajar tidak mampu untuk menyelesaikan masalah atau menjawab soalan yang diberikan walaupun kebanyakan soalan adalah ulangan latihan di dalam kelas dengan sedikit pengubahsuaian. Pengubahsuaian tersebut dilakukan untuk dapat menguji kemahiran pelajar untuk berfikir secara kreatif. Akhirnya, ini akan menyebabkan prestasi pelajar dinilai pada tahap yang tidak memuaskan dan mengakibatkan kadar kegagalan yang tinggi tinggi. Daripada 78,329 pelajar yang telah enrol ke dalam bidang Sains, Matematik dan Komputer di IPTA, hanya 16,789 pelajar, iaitu 21% sahaja merupakan hasil keluaran dari bidang tersebut. (Kementerian Pendidikan Tinggi Malaysia 2016.)

Kurangnya latihan pengaturcaraan yang dilakukan oleh pelajar dianggap sebagai punca utama kurangnya kemahiran seseorang pelajar dalam pengaturcaraan. Malah, proses pembelajaran dan pengajaran tradisional yang masih lagi digunakan untuk kursus pengaturcaraan di IPTA juga menjadi salah satu punca. Proses tradisional tersebut tidak dapat menarik minat pelajar agar menjadi lebih bermotivasi untuk mempelajari dan menguasai pengaturcaraan. Sejarar dengan itu, pelbagai kajian untuk menggabungkan elemen-elemen permainan dalam pendidikan telah dilakukan kerana kebanyakan pelajar dapat mempelajari sesuatu perkara dengan lebih baik apabila kita bermotivasi tinggi. (Bergin dan Reilly 2005).

Permainan komputer menjadi satu fenomena yang mampu menarik perhatian dan minat pelajar generasi kini. Mereka akan terus bermain permainan komputer sehingga mereka dapat

mencapai satu matlamat, iaitu pengakhiran dan kejayaan mengakhiri permainan tersebut. Pendedahan kepada permainan komputer dengan secukupnya membuatkan mereka tertarik untuk mempelajari tentang komputer, terutamanya pengaturcaraan (Burguillo 2010). Permainan komputer sebagai medium pengajaran dan pembelajaran mempunyai nilai intrinsiknya yang tersendiri untuk meningkatkan sifat ingin tahu seseorang pelajar (Kumar 2000). Menurut kajian yang telah dilakukan oleh Ip, Jacob dan Watkins (2008), 88% daripada sampel pelajar dari Universiti Swansea pernah bermain permainan video, malah 71% daripada sampel tersebut kerap bermain permainan video (sekurang-kurangnya satu jam sehari). Permainan video atau permainan komputer mampu memotivasikan pemain kerana pemain boleh mengetahui sejauh mana kemajuan mereka dan setiap kejadian serta tindakan yang dilakukan akan membuahkan hasil yang dapat dilihat dengan serta merta.

Justeru itu, pendekatan baharu yang lebih menarik dan efektif serta sistematik diperlukan untuk mengatasi masalah yang dihadapi oleh pelajar dan pensyarah. Pendekatan yang dicadangkan ini menerapkan unsur-unsur gamifikasi dan mengintegrasikan Pembelajaran Berasaskan Pertandingan untuk mendorong dan menarik minat pelajar untuk menguasai pengaturcaraan dengan sebaiknya. Di samping itu pendekatan ini dapat memupuk semangat berdaya saing antara pelajar. Usulan ini mencadangkan pembinaan suatu persekitaran pembelajaran pengaturcaraan yang kompetitif dan interaktif, di mana setiap aktiviti yang dilakukan oleh pelajar akan diambil kira ke dalam sistem pemarkahan. Penilaian secara automatik memberi maklumbalas segera yang membolehkan pelajar terus mencuba latihan berikut atau membaiki kesilapan latihan yang diberi. Proses pemarkahan dan penilaian secara automatik ini diperkenalkan sebagai insentif untuk memberikan pelajar suatu kepuasan dalam menyiapkan sesuatu latihan atau tugas. Di samping itu dapat meringankan beban pensyarah untuk menilai pelajar. Pensyarah boleh memberi tumpuan untuk membantu pelajar yang lemah.

2 PENYATAAN MASALAH

Fakulti Teknologi dan Sains Maklumat (FTSM) di Universiti Kebangsaan Malaysia (UKM) telah menggunakan sistem Programming Contest Control (PC²) sebagai perisian utama untuk menjalankan sesi makmal dan ujian makmal. Sesi makmal adalah satu sesi tutorial di mana pelajar akan menyiapkan tugas atau latihan yang diberikan oleh pensyarah di Makmal Pengaturcaraan dengan panduan yang diberikan oleh pensyarah. Ujian makmal pula ialah satu

ujian atau peperiksaan yang dilakukan untuk menguji kemahiran pelajar setelah menjalani beberapa sesi tutorial makmal.

Pertandingan pengaturcaraan komputer ialah satu pertandingan yang mana peserta perlu menyelesaikan masalah yang diberikan dengan menghantar jawapan mereka (iaitu program komputer/kod sumber pengaturcaraan mereka) kepada juri untuk disemak dan dinilai. PC² merupakan satu sistem terbuka yang paling popular dan sering digunakan untuk pertandingan pengaturcaraan di seluruh dunia. Walaupun penggunaan PC² untuk sesi makmal memberi impak yang positif, namun disebabkan ianya dibina sebagai sistem pengurusan pertandingan pengaturcaraan, PC² telah menimbulkan beberapa masalah.

Antaranya masalah penggunaan PC² dalam kursus Pengaturcaraan Komputer di FTSM ialah pelayan PC² FTSM hanya boleh dicapai di dalam Makmal Pengaturcaraan di FTSM. Ini menyebabkan PC² kurang sesuai untuk dijadikan sebagai sistem untuk pembelajaran pengaturcaraan sendiri secara berterusan. Pelajar tidak berpeluang untuk meneruskan pengaturcaraan dan mencari jawapan kepada soalan atau masalah yang diberikan oleh pensyarah di luar masa makmal. Ini mengakibatkan pelajar kurang bersemangat untuk menyiapkan tugas mereka kerana mereka tidak boleh menghantarnya dengan serta merta walaupun mereka telah menyiapkan aturcara mereka. Malah, mereka juga tidak berpeluang untuk mengetahui sama ada aturcara yang mereka telah tulis mampu untuk menyelesaikan masalah yang diberi ataupun tidak.

Untuk memudahkan para juri (iaitu pensyarah pengaturcaraan FTSM dalam kes ini), PC² juga mempunyai sistem penilaian automatik yang lebih dikenali sebagai Automatic Judge (Juri Automatik). Juri Automatik ini akan mengkompil dan melaksanakan setiap aturcara tersebut. Kemudian, ia akan semak output yang dihasilkan oleh program yang diserahkan oleh peserta itu sama ada ianya betul dan tepat berbanding dengan output juri. Di sini timbulnya masalah kedua. Walaupun PC² mempunyai sistem penilaian automatik, ianya tidak lengkap dan kurang sesuai untuk menggantikan pensyarah sepenuhnya untuk menilai prestasi pelajar. Hal ini kerana PC² hanya memberikan markah secara nombor binar. 0 - Wrong Answer/Error dan 1 - Accepted. Sistem pemarkahan tersebut kurang sesuai untuk kegunaan di dalam proses pengajaran dan pembelajaran kerana aturcara yang ditulis perlu melepasi semua kes ujian untuk mendapatkan 1 markah tersebut.

PC² tidak mempunyai bank soalan. Ini menyukarkan pensyarah untuk menyusun atau mengklasifikasikan jenis dan tahap latihan atau tugas yang ingin diberikan kepada pelajar. PC² juga tidak mempunyai pangkalan data untuk menyimpan maklumat pelajar. Pensyarah perlu menyimpan segala maklumat prestasi pelajar secara manual. Ini menyebabkan pensyarah sukar untuk melihat prestasi setiap pelajar dan menyemak tugas-tugas yang telah dihantar mengikut kelas-kelas yang tersendiri dengan sistematik dan teratur.

3 OBJEKTIF KAJIAN

Objektif projek ini adalah untuk membina sistem pembelajaran bersepadu sebagai medium utama pengajaran dan pembelajaran pengaturcaraan. Objektif projek ini akan tercapai dengan melalui beberapa objektif khusus, seperti membangunkan sistem penilaian automatik untuk latihan pengaturcaraan dan mengintegrasikan persekitaran kompetitif dalam sistem yang dibangunkan.

4 METOD KAJIAN

Sepanjang pembangunan sistem dalam projek ini, metodologi yang digunakan ialah eXtreme Programming (XP), iaitu salah satu kitaran hidup pembangunan perisian yang berjenis Agile. Terdapat lima fasa dalam XP termasuklah Perancangan, Rekabentuk, Pengekodan, Pengujian dan Maklumbalas.

4.1 Fasa Perancangan

Fasa Perancangan bertujuan menentukan matlamat dan objektif keseluruhan projek ini serta kitaran lelaran (*iteration cycles*) yang tertentu. Perjumpaan bersama penyelia akan diadakan untuk mencipta 'Cerita Pengguna' (*User Stories*) atau kajian keperluan untuk memastikan projek ini berjalan dengan lancar. 'Cerita Pengguna' yang ada kemudiannya akan dijadikan sebagai lelaran yang merangkumi sebahagian kecil daripada fungsi yang diperlukan dan beberapa lelaran akan digabungkan supaya terbentuknya satu produk yang lengkap dan berfungsi. Fasa ini juga melibatkan penganggaran masa dan kos serta menentukan keutamaan lelaran yang sepatutnya.

4.2 Fasa Rekabentuk

Fasa Rekabentuk merujuk kepada pentakrifan fungsi utama kod pada masa akan datang. Setiap lelaran akan dimulakan dengan mereka bentuk. Menurut metodologi XP, sesebuah rekabentuk perlulah ringkas. Oleh itu, rekabentuk bagi antaramuka sistem ini bukan sahaja ringkas malah berfungsi dengan baik dan kemas serta seragam dengan penampilan dan persekitaran di FTSM, UKM. Gambar rajah mendemonstrasikan antaramuka sistem akan turut diwujudkan untuk memastikan interaksi antaramuka pengguna dan aliran sistem ini dapat ditunjukkan. Selain itu, bahagian belakang sistem ini termasuklah aliran juga akan direkabentuk dengan berhati-hati untuk memastikan segala fungsi dan kod akan berfungsi dengan baik dan ringkas, seperti memilih platform yang bersesuaian dengan skop projek dan memastikan kerangka yang dipilih bersesuaian dengan matlamat sistem ini.

4.3 Fasa Pengekoden

Fasa Pengekoden adalah fasa mengekod sistem di mana segala perkara yang telah ditetapkan dalam Fasa Rekabentuk akan dihasilkan sebagai kod yang sebenar. Fasa Pengekoden akan dilaksanakan terus setelah Fasa Rekabentuk selesai dan akan proses yang sama akan berulang untuk setiap lelaran sehingga sistem ini siap dibangunkan. Pengubahsuaian kepada projek, sekiranya perlu, akan dirujuk semula pada Fasa Rekabentuk dan kitaran lelaran yang kini akan diubahsuai kemudian diulang semula supaya dapat menghasilkan produk dan mencapai matlamat yang sepatutnya (bagi setiap lelaran).

4.4 Fasa Pengujian

Fasa Pengujian dilakukan secara berterusan semasa Fasa Pengekoden sedang dijalankan. Pengujian akan dijalankan setiap kali sesuatu kelas atau modul telah disiapkan untuk menghapuskan segala pepijat (*bugs*) dan semua kod perlu berjaya melepasi Fasa Pengujian sebelum produk dihasilkan untuk fasa seterusnya.

4.5 Fasa Maklumbalas

Fasa Maklumbalas dijalankan bersama penyelia. Penyelia akan memberikan maklum balas dan pendapat beliau tentang produk yang dihasilkan. Asas kepada maklumbalas tersebut merupakan pengujian sambutan atau penerimaan penyelia terhadap produk tersebut. Setiap maklumbalas yang diberikan untuk produk yang dihasilkan dalam sesebuah lelaran akan diambilkira sebagai rekabentuk yang baharu dan kemudian lelaran tersebut akan diulang semula. Sekiranya produk yang dihasilkan untuk lelaran tersebut memuaskan, lelaran ini akan

berakhir dan lelaran seterusnya dimulakan, lantas proses setiap fasa akan berulang dengan lelaran yang baharu.

5 HASIL KAJIAN

Sejajar dengan metodologi eXtreme Programming yang dipilih untuk membangunkan sistem ini, proses mengimplementasi dan menguji sistem dilakukan secara pertambahan (*incrementally*), juga dikenali sebagai Incremental Build Model. Sistem CodeRoom merupakan sebuah sistem yang berasaskan web. Oleh itu, pelbagai teknologi telah dipilih untuk digunakan dalam pembangunan sistem ini.

Untuk pembangunan *backend* sistem CodeRoom, PHP dipilih sebagai bahasa pengaturcaraan utama. SQL juga digunakan untuk penyimpanan dan memanipulasi data di dalam pangkalan data MySQL. Manakala pembangunana *frontend* (antaramuka) pula, gabungan antara HTML, CSS dan JavaScript digunakan untuk membangunkan antaramuka yang kemas dan menarik. Sistem ini dibangunkan menggunakan *Visual Studio Code* sebagai editor kod secara keseluruhannya. *Visual Studio Code* menyokong pelbagai bahasa termasuklah PHP, HTML, CSS dan JavaScript. Rajah 5.1 menunjukkan antara muka *Visual Studio Code*.

The screenshot displays the Visual Studio Code interface. The main editor shows the PHP file `ProcessSubmission.php` with the following code:

```

41 ...../**
42 ..... * Execute the job.
43 ..... */
44 ..... * @return void
45 ..... */
46 ..... public function handle()
47 ..... {
48 .....     $msg = 'Processing new job...asdasd'. "\n";
49 .....
50 .....     $user = $this->getUser($this->submission->submitterId);
51 .....     $problem = $this->getProblem($this->submission->problemId);
52 .....     $language = $this->getLanguage($this->submission->languageId);
53 .....
54 .....     $msg .= 'Submission ID: '.$this->submission->id. "\n";
55 .....     $msg .= 'User: '.$user->name.' (ID: '.$user->id.'). "\n";
56 .....     $msg .= 'Language: '.$language->name. "\n";
57 .....
58 .....     if($this->submission->classId !== NULL) {
59 .....         $classroom = $this->getClassroom($this->submission->classId);
60 .....         $msg .= 'Classroom ID: '.$classroom->id. "\n";
61 .....     }
62 ..... }

```

The terminal window at the bottom shows the following output:

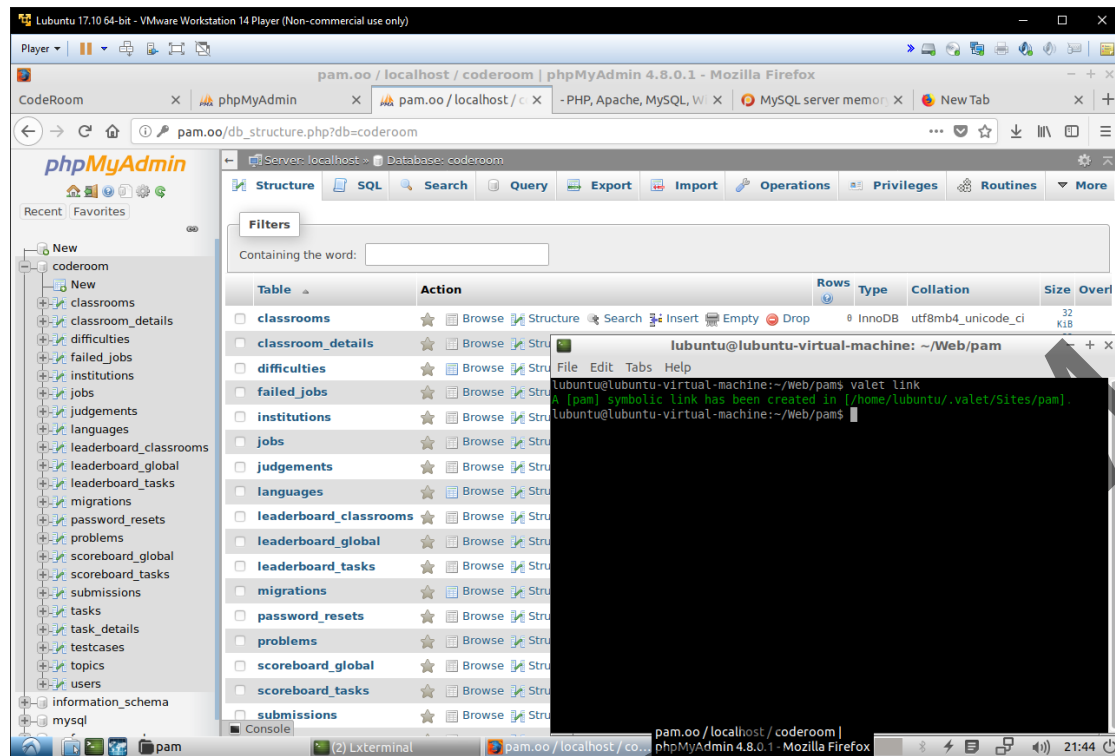
```

Migrating: 2018_05_09_173100_create_scoreboard_tasks_table
Migrated: 2018_05_09_173100_create_scoreboard_tasks_table
Migrating: 2018_05_09_180713_create_leaderboard_globals_table
Migrated: 2018_05_09_180713_create_leaderboard_globals_table
Migrating: 2018_05_09_181040_create_leaderboard_tasks_table
Migrated: 2018_05_09_181040_create_leaderboard_tasks_table
Migrating: 2018_05_09_181108_create_leaderboard_classrooms_table
Migrated: 2018_05_09_181108_create_leaderboard_classrooms_table
Seeding: UsersTableSeeder
Seeding: DifficultiesTableSeeder
Seeding: LanguagesTableSeeder
PS C:\laragon\www\CodeRoom> /c:/laragon/www/CodeRoom/app/Jobs/ProcessSubmission.php

```

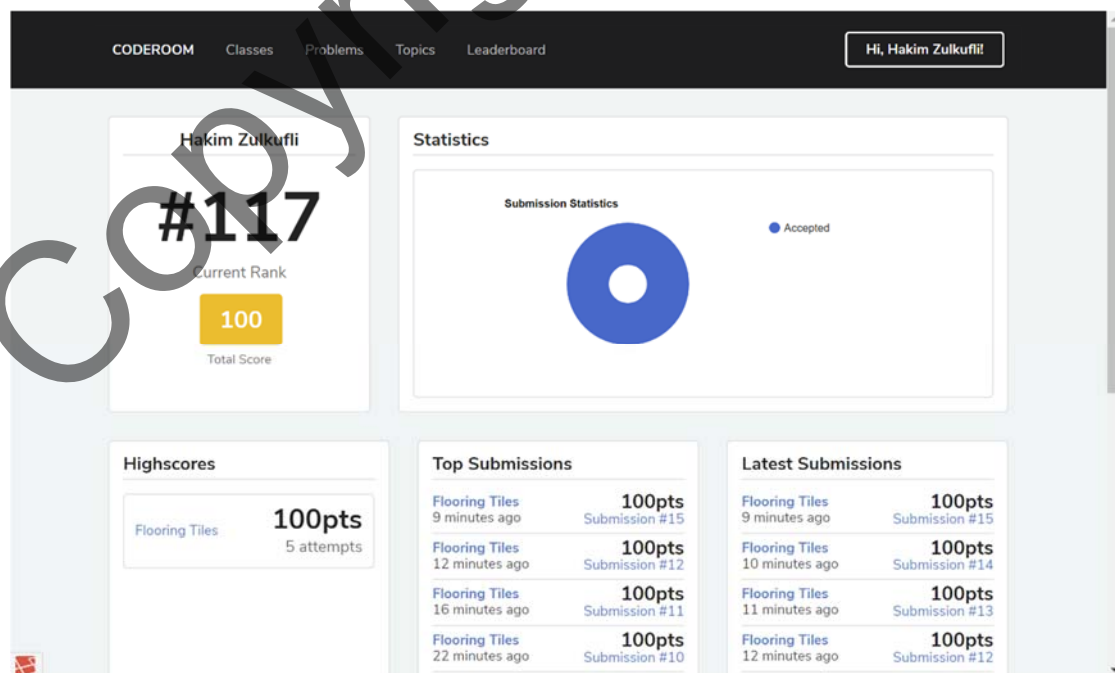
Rajah 5.1 Antara muka *Visual Studio Code*.

Walaupun sistem ini dibangunkan di dalam sistem pengoperasian *Microsoft Windows*, sistem ini disasarkan untuk digunakan di dalam pelayan yang menggunakan sistem pengoperasian *Linux*. Untuk memudahkan kerja-kerja pengujian dan *debugging*, sebuah mesin maya turut digunakan, iaitu *VMware Workstation 14 Player*. Rajah 5.3 menunjukkan antara muka *VMware Workstation 14 Player*.



Rajah 5.2 Antara muka VMWare Workstation 14 Player.

CodeRoom adalah sebuah sistem yang berasaskan web. Oleh itu, segala fungsi yang ingin digunakan oleh pengguna perlu dilakukan secara atas talian melalui pelayar internet masing-masing. Rajah 5.3 hingga 5.11 menunjukkan antara muka penting yang dipaparkan untuk menggunakan fungsi teras yang terdapat dalam sistem CodeRoom.



Rajah 5.3 Paparan profil pengguna.

The screenshot shows the CODEROOM interface with the 'Leaderboard' tab selected. The page title is 'Leaderboard' and the user is logged in as 'Hi, Hakim Zulkuffi!'. The table displays the following data:

#	Name	Total Score
1	Gideon McGlynn	2038
2	Shaylee Jacobs	2015
3	Jacques Hudson	2011
4	Prof. Ashlee Koss	2006
5	Mr. Taylor Schmitt II	2002
6	Kirk Macejkovic	1993
7	Beverly Kunze	1988
8	Nestor Feest IV	1954
9	Prof. Hosea Schimmel	1948
10	Madeline Weber	1942
11	Mr. Noe Murray	1936
12	Opal Wolff I	1928
13	Alyson Anderson	1890
14	David Stokes	1870
15	Marilia Donlar	1847

Rajah 5.4 Paparan senarai kedudukan pengguna *global*.

The screenshot shows the CODEROOM interface for the class 'Pengaturcaraan Komputer'. The page title is 'Pengaturcaraan Komputer' and the user is logged in as 'Hi, Hakim Zulkuffi!'. The class description is 'Belajar asas kemahiran pengaturcaraan menggunakan Java'. The 'Leaderboard' tab is selected, showing the following data:

#	Name	Total Score	Global	
			Total Score	Rank
1	A123455 Hakim Zulkuffi	100	100	117
2	A164985 Signe Cormier	0	1733	25
3	A652881 Stacey Weber DVM	0	263	106
4	A696681 Domingo Walker	0	1205	53
5	A686748 Chloia Kutch	0	1817	17
6	A690588 Eric Macejkovic MD	0	1004	66
7	A172479 Dr. Iliana Conn V	0	78	119
8	A478758 Meagan Ritchie V	0	1338	43
9	A814433 Levi Hahn	0	198	110
10	A104793 Prof. Ray Skiles	0	1158	55
11	A194868 Katelyn Buckridge	0	570	92

Rajah 5.5 Paparan kedudukan pelajar di dalam kelas.

Pengaturcaraan Komputer
Belajar asas kemahiran pengaturcaraan menggunakan Java.

Leaderboard Tasks Info

Tutorial 1
Do this tutorial and you shall get A.

Problem

- Flooring Tiles
- Ben 10+
- My Date
- Which Quadrant?
- Secret Message

Faculty of Information Science & Technology
Universiti Kebangsaan Malaysia
43600 Bangi, Selangor
Malaysia

Rajah 5.6 Paparan senarai soalan dalam set latihan.

Flooring Tiles
Arithmetic Expression

The faculty is initiating a drive to upgrade the facilities of the computer laboratory. Your group is assign to be in charge of the flooring.

Your group task is to determine the dimension of all labs involved. You are also required to write a program that calculates the number of tiles needed for each lab. The size of the proposed tile is 30cm x 30cm. Note that there might be a row and a column of tiles that need to be cut so that it can be lay perfectly.

Input

The first line of input is the number of labs n , where $1 \leq n \leq 20$. For each of the following n lines there are 2 integers, width (cm) ($1 \leq \text{width} \leq 10,000$) and length (cm) ($1 \leq \text{length} \leq 10,000$) which represent the dimension of the floor for the respective lab.

Output

For each lab, print the number of tiles needed.

Sample Input

1	3
2	300 300
3	500 1000
4	1000 900

Sample Output

1	100
2	578
3	1020

Info

CPU Time Limit
1 seconds

Memory Limit
10 MB

Difficulty
Easy

Top Scorers

#1	Hakim Zulkuffi	100pts 5 attempts (100pts)
----	----------------	--------------------------------------

Rajah 5.7 Paparan soalan.

The screenshot shows the 'Create Class' form in the CODEROOM application. The user is logged in as 'Hi, Abdul Manaf!'. The form has the following fields and options:

- Code:** TK1114
- Name:** Pengaturcaraan Komputer
- Description:** Basics of programming with Java.
- Status:** Active Inactive
- Buttons:** Create (green), Cancel (grey)

The footer of the page includes the Faculty of Information Science & Technology logo and contact information for Universiti Kebangsaan Malaysia, along with a note 'Coded with ❤️ by Hakim Zulkuffli'.

Rajah 5.8 Paparan mencipta kelas.

The screenshot shows the 'My Topics' page in the CODEROOM application. The user is logged in as 'Hi, Zheng Yao!'. The page displays a list of topics with 'Edit' and 'Delete' buttons for each:

Name	Edit	Delete
Arithmetic Expression	Edit	Delete
Repetition Structure	Edit	Delete
Conditional Structure	Edit	Delete
String Processing	Edit	Delete

Below the list is a '← All Topics' button. At the bottom, there is a 'Create Topic' form with a 'Name' field containing 'Problem Solving' and a 'Create Topic' button.

Rajah 5.9 Paparan menguruskan topik.

EDIT PROBLEM

Info Testcases

Name: Flooring Tiles Code: AE1111

Difficulty: Easy Topic: Arithmetic Expression Time Limit: 1 secs Memory Limit: 10 MB

Status: Active Inactive Hide from Students: No Yes

Problem Statement

The faculty is initiating a drive to upgrade the facilities of the computer laboratory. Your group is assign to be in charge of the flooring.

Your group task is to determine the dimension of all labs involved. You are also required to write a program that calculates the number of tiles needed for each lab. The size of the proposed tile is 30cm x 30cm. Note that there might be a row and a column of tiles that need to be cut so that it can be lay perfectly.

Rajah 5.10 Paparan memasukkan maklumat soalan.

Languages

Name: Java [Edit] [Delete]

← Back

Edit Language

Name: Java

Source Code File Extension: java

Source Code Filename: Main.java

Compiled Executable Filename: Main.class

Compile Command: ./usr/bin/javac [PATH]/Main.java

Rajah 5.11 Paparan menguruskan bahasa pengaturcaraan.

6 KESIMPULAN

Sistem CodeRoom merupakan satu sistem pembelajaran yang berasaskan laman sesawang untuk memudahkan proses pengajaran dan pembelajaran pengaturcaraan dalam sesebuah kelas. CodeRoom mempunyai penilaian automatik yang akan menjalankan kod sumber yang dihantar

oleh pelajar dan memberikan markah secara automatik. Sistem ini membolehkan pelajar terutamanya untuk menjawab dan menyelesaikan masalah pengaturcaraan secara berterusan tanpa memerlukan pelajar perlu bersama atau bertanyakan kepada pensyarah untuk mendapatkan markah atau mengetahui sama ada jawapan yang ditulis adalah betul atau tidak. Pensyarah juga tidak perlu untuk menjalankan pengaturcaraan setiap pelajar dan memberikan markah secara manual.

Sistem yang dibangunkan ini berjaya mencapai objektif-objektif yang telah dinyatakan bagi menyelesaikan masalah dalam kelas pengaturcaraan di universiti. Sistem yang dibangunkan diharap dapat membantu pensyarah dan pelajar dalam menjalankan proses pengajaran dan pembelajaran kelas pengaturcaraan dengan lebih lancar terutamanya di UKM.

7 RUJUKAN

- Aleryani, A. 2016. *Comparative Study between Data Flow Diagram and Use Case Diagram. International Journal of Scientific and Research Publications*, hlm.Vol. 6.
- Basnet, R. B., Doleck, T., Lemay, D. J. & Bazelais, P. 2017. Exploring computer science students' continuance intentions to use Kattis. *Education and Information Technologies*,. doi:10.1007/s10639-017-9658-2
- Benford, S., Burke, E. & Foxley, E. 2004. *Courseware to support the teaching of programming*. University of Nottingham. Retrieved from <http://pessoal.sercomtel.com.br/assis/C/CEILIDH/Courseware.cat>
- Bergin, S. & Reilly, R. 2005. The influence of motivation and comfort-level on learning to program. *Ppig 17*, (June), 293–304. Retrieved from <http://www.cs.nuim.ie>
- Bloomfield, A. & Sotomayor, B. 2016. A Programming Contest Strategy Guide. *Proceedings of the 47th ACM Technical Symposium on Computing Science Education, SIGCSE '16* hlm.609–614. New York, NY, USA: ACM. doi:10.1145/2839509.2844632
- Burguillo, J. C. 2010. Using game theory and Competition-based Learning to stimulate student motivation and performance. *Computers and Education*, 55(2), 566–575. doi:10.1016/j.compedu.2010.02.018
- California State University Sacramento. 2015. Judge's Guide ACM INTERNATIONAL COLLEGIATE PROGRAMMING CONTEST. Retrieved from <http://pc2.ecs.csus.edu/>
- California State University Sacramento. (n.d.). Contest Administrator's Installation and Configuration Guide. Retrieved from <https://pc2.ecs.csus.edu/doc/v9/9.5.2/pc2v9AdminGuide.pdf>

- California State University Sacramento. (n.d.). CSUS Programming Contest Control (PC²). <https://pc2.ecs.csus.edu/> [10 December 2017].
- California State University Sacramento. (n.d.). PC2 Version 9 Contestant's Guide. <https://pc2.ecs.csus.edu/doc/v9/9.5.2/PC2V9TeamGuide.pdf> [10 December 2017].
- Ip, B., Jacobs, G. & Watkins, A. 2008. Gaming frequency and academic performance. *Australasian Journal of Educational Technology*, 24(4), 355–373. doi:10.14742/ajet.1197
- Kumar, D. 1999. Curriculum Descant: Pedagogical Dimensions of Game Playing. *Intelligence*, 10(1), 9–10. doi:10.1145/298475.298480
- Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., McCartney, R. et al. 2004. A Multi-National Study of Reading and Tracing Skills in Novice Programmers. *SIGCSE Bulletin*, 36(4), 119–150. Retrieved from <https://opus.lib.uts.edu.au/bitstream/10453/4126/3/2004000904.pdf>
- Ministry of Higher Education. 2016. Public Universities. *Higher Education Statistics 2015*, 21–54. Retrieved from <https://www.mohe.gov.my/ms/muat-turun/awam/statistik/2015/217-bab-2-universiti-awam-1/file>
- Moström, J. E. 2011. *A study of Student Problems in Learning to Program*. doi:978-91-7459-293-1
- Njoku, D., Nkuma-Udah, K. . & Onwugbufo, C. G. 2013. Client – Server Architecture: An Overview 58–65. Retrieved from http://www.africanbmes.org/ajmbs/njoku_fullpaper_4_2.pdf
- Oracle. (n.d.). The Data Dictionary. https://docs.oracle.com/cd/B28359_01/server.111/b28318/datadict.htm [10 December 2017].
- Petit, J., Giménez, O. & Roura, S. 2012. *Jutge.org: an educational programming judge*. doi:10.1145/2157136.2157267
- Petit, J., Roura, S., Carmona, J., Cortadella, J., Duch, A., Gimenez, O., Mani, A. et al. 2017. *Jutge.org: Characteristics and Experiences*. *IEEE Transactions on Learning Technologies*, hlm. Vol. PP. doi:10.1109/TLT.2017.2723389
- The Writing Center. (n.d.). Literature Reviews. <https://writingcenter.unc.edu/tips-and-tools/literature-reviews/> [10 December 2017].
- Zin, A. M. & Foxley, E. 1991. *Automatic Program Quality Assessment System*. University of Nottingham. Retrieved from <http://www.cs.nott.ac.uk/Department/Staff/ef/ceilidh/papers/ASQA.html>