Pentru a stăpâni limbaje de programare nu este de ajuns doar înțelegerea gramaticii limbajului și a instrumentelor de programare. Este indispensabilă efectuarea în mod repetat a multor exerciții practice de programare.

[la finalul descrierii sistemului]

Trebuie remarcat faptul că această lucrare prezintă cazul limbajului C, dar abordarea noastră poate fi extinsă cu ușurință la alte limbaje de programare. Sistemul nostru prototip este situat la <http://pplinux.is.konan-u.ac.jp/rccE.html>.

Prin urmare, deoarece cursanții nu vor observa că efectuează în mod repetat programe generate din același șablon, sistemul nostru este potrivit pentru învățarea repetitivă. Trebuie remarcat faptul că programele șablon sunt pregătite în conformitate cu progresul învățării și sunt necesare aproximativ 2, 3 de programe șablon pentru a face față unei varietăți de programe.

Trebuie remarcat faptul că abordarea noastră cu același script PHP este ușor extinsă la alte limbaje de programare prin schimbarea compilatoarelor și rescrierea programelor șablon și a matricelor sp și rp.

După cum s-a menționat mai sus, parametrii constanti și modelele de calcul ale programelor trebuie să fie date corect, deoarece parametrii greșiți pot provoca excepții de diviziune la zero și erori de acces la memorie.

[rezultate experimentale]

De asemenea, confirmăm eficacitatea sistemului prin experimentele noastre preliminare.

eficacitatea practicilor de depanare este confirmată empiric. În viitor, vom evalua următoarele aspecte: • evaluarea practicilor de estimare a rezultatelor, • combinația dintre sistemul nostru și MOOC, • exercițiile în conformitate cu progresul învățării și • evaluarea profundă de către mulți evaluatori. Mai mult, ar trebui să implementăm și sistemul practic și să-l evaluăm

În cadrul lucrării, autorii au pornit de la premiza că la baza învățării unui limbaj de programare se află atât dobândirea cunoștințelor de gramatică a limbajului cât și capabilitatea de a înțelege fluxul unui program. Prin urmare, aceștia au pregătit două tipuri de exerciții: exerciții de sintaxă prin care cursanții să corecteze erorile programelor furnizate, și exerciții de semantică prin care cursanții să estimeze rezultatul programelor, fără a le executa. Acestea pot fi accesate prin intermediul unei aplicații web în limbajul PHP, care le generează automat folosind șabloane în limbajul C. Exercițiile generate sunt evaluate automat.

Pentru primul tip de exerciții, interfața grafică permite vizualizarea unui program cu erori și a descrierilor erorilor furnizate de compilator. Prin intermediul acesteia, cursanții corectează programul utilizând cunoștințele despre gramatica limbajului și mesajele de eroare până când toate erorile sunt complet eliminate. Exercițiile sunt generate prin adăugarea de bug-uri unor programe șablon și prin înlocuirea numelor de variabile.

Au fost definite trei metode de adăugare a erorilor la programul șablon. În primul rând, numele variabilelor sunt înlocuite cu valori alese aleator dintr-o listă predefinită de nume. Astfel, pot apărea în program variabile nedeclarate. A doua metodă se bazează pe scrierea greșită a unor șiruri de caractere definite în prealabil (de exemplu, stdio.h poate fi schimbat în stdi.h). A treia metodă înlocuiește un șir de caractere cu un alt cuvânt desemnat, pentru a genera o eroare întâlnită des (de exemplu, înlocuirea „;” cu „:” sau „float” cu „single”).

După ce un cursant corectează un program care are erori, scriptul PHP salvează programul corectat ca fișier cu un nume unic, îl compilează și trimite înapoi mesajul de la compilator către cursant. Șirul de caractere „OK” este trimis înapoi, dacă compilatorul nu produce niciun mesaj.

Autorii menționează că programele șablon sunt pregătite în conformitate cu progresul învățării și sunt necesare între 10 și 20 programe șablon pentru a produce o varietate de programe.

Al doilea tip de exerciții, cele de estimare a output-ului unui program, au scopul de a îmbunătăți capacitatea de înțelegere a programelor din punct de vedere semantic. Interfața grafică a aplicației web permite vizualizarea unui program care conține o instrucțiune de ieșire. Cursantul citește programul și estimează rezultatul funcției printf prin înțelegerea semanticii acestuia. După scrierea valorii estimate în zona de răspuns și apăsarea butonului de verificare, este afișată comparația răspunsului corect cu estimarea.

Autorii menționează că programele date pot include apeluri recursive de funcții și calcule complicate, ceea ce face mai dificil de estimat rezultatul. Pentru a crea programe corecte, este indispensabilă înțelegerea comportamentului programelor care includ instrucțiuni condiționale, bucle și apeluri de funcții. Prin urmare, aceștia susțin că aceste exerciții sunt utile pentru a îmbunătăți capacitatea cursanților de a înțelege programe. Prin schimbarea adecvată a numelor variabilelor și a valorilor constantelor și menținând lungimea programului suficient de compactă pentru estimare, cursanții pot efectua exerciții în mod repetat pentru a spori eficacitatea educației. Parametrii constanți și modelele de calcul trebuie să fie date corect, deoarece parametrii greșiți pot provoca excepții precum împărțirea la zero sau erori de acces la memorie.

NU SUNT DE ACORD

Prin urmare, deoarece cursanții nu vor observa că efectuează în mod repetat programe generate din același șablon, sistemul nostru este potrivit pentru învățarea repetitivă.