

Logica Propozițională. Seminar 1 - Fișă de exerciții (30.09.2019 - 04.10.2019)

Unele exerciții utilizate mai jos se regăsesc și în cartea *Peter Smith. An Introduction to Formal Logic. Cambridge University Press, 2003.*

Propoziții. O propoziție este un enunț care poate fi adevărat sau fals. Mai jos sunt câteva exemple de propoziții:

1. “Tu ai un laptop.” 2. “Zăpada este albă.” 3. “Zăpada nu este albă.” 4. “Tatăl meu merge la servicii și eu merg la școală.” 5. “Afară plouă, dar eu am umbrelă.” 6. “Măine va ploua sau nu va ploua.” 7. “Dacă obțin notă de trecere la logică, voi sărbători.” 8. “ $2 + 2 = 4$.” (“Doi plus doi egal cu 4.”)

```
int suma(int n)
{
    int i    = 0;
    int sum = 0;
    while (i < n)
    {
        i    = i    + 1;
        sum = sum + i;
    }
    return sum;
}
```

Figure 1: Programul `suma`.

Exercițiul 1. Fie programul `suma` din Figura 1. Următoarele propoziții sunt adevărate sau false: 1. “Programul `suma` calculează suma primelor n numere naturale.” 2. “Dacă interschimbăm instrucțiunile din corpul buclei `while`, atunci `suma` nu calculează suma primelor n numere naturale.” 3. “Dacă valoarea de intrare pentru n este mai mică decât 0 atunci `suma` returnează 0.”

Următoarele exemple nu sunt propoziții deoarece ele nu sunt nici adevărate, nici false:

- “Roșu și Negru.” (nu este enunț)
- “ π .” (nu este enunț)
- “Plouă?” (întrebare)
- “Hai la pescuit!” (exclamație)
- “ x este mai mare decât 7.” (nu putem ști până nu îl cunoaștem pe x)
- “Această afirmație este falsă.”

Exercițiul 2. Puteți formula un enunț despre programul `sum` care să nu fie propoziție?

Raționamente. Un raționament este o secvență de propoziții. Ultima propoziție se numește *concluzia* raționamentului, iar celelalte propoziții (mai puțin ultima) se numesc *premize*. De obicei, concluzia este precedată de cuvinte precum “așadar”, “deci”, “prin urmare”. Mai jos avem exemple de raționamente:

A1:

1. “Ionel aleargă mai repede decât Maria.”
2. “Maria aleargă mai repede decât Mihai.”
3. Prin urmare, “Ionel aleargă mai repede decât Mihai.”

A2:

1. “Toți oamenii sunt muritori.”
2. “Socrates este om.”

3. Așadar, “Socrates este muritor.”

Exercițiul 3. *Răspundeți la următoarele întrebări:*

- Care sunt premisele și concluziile pentru A1? Dar pentru A2?
- Vi se par corecte/convingătoare aceste raționamente?

Putem construi argumente care nu sunt neapărat corecte sau convingătoare:

1. “Toți studenții sunt inteligenți.”
2. “Ionel este o persoană.”
3. Deci “Pământul este rotund.”

Cum ne putem da seama că un raționament este corect? Există două condiții pe care raționamentele trebuie să le îndeplinească pentru a fi corecte:

1. Concluzia trebuie să fie o consecință a premizelor raționamentului.
2. Premizele raționamentului sunt adevărate.

Orice raționament care satisface prima condiție se numește raționament *deductiv valid* (engleză: valid sau deductively valid). Orice raționament care satisface ambele condiții este *corect* (sound).

Exercițiul 4. *Raționamentele de mai jos sunt raționamente (deductiv) valide? Dar corecte?*

- Ionel este mai înalt decât Maria. Ioana este mai scundă decât Maria. Deci, Ionel este mai înalt decât Ioana.
- Coca-cola este un suc. Toate sucurile sunt lichide. Coca-cola este un lichid.
- Unii točilari sunt pasionați de trenuri. Unii točilari poartă pelerine de ploaie. Deci, unii točilari pasionați de trenuri poartă pelerine de ploaie.
- Mulți politicieni iau mită. Cei mai mulți politicieni au afaceri ilegale. Așadar, mulți oameni care iau mită au afaceri ilegale.
- Ionel joacă fotbal sau baschet. Deci, dacă Ionel nu joacă fotbal atunci el joacă baschet.
- Toți studenții sunt inteligenți. Unii studenți sunt leneși. Prin urmare, unii studenți leneși sunt inteligenți.
- Nici un student nu este leneș. Nici un student leneș nu se trezește dimineața. Așadar, unii studenți nu se trezesc dimineața.
- Valoarea de intrare pentru n în programul `suma` este mai mică decât 0. Dacă valoarea de intrare pentru n în programul `suma` este mai mică decât 0, atunci valoarea întoarsă de programul `suma` este 0. Prin urmare, valoarea întoarsă de programul `suma` este 0.
- Valoarea de intrare pentru n în programul `suma` este 1. Dacă valoarea de intrare pentru n în programul `suma` este 1 atunci corpul buclei este executat o singură dată. Dacă corpul buclei este executat o singură dată atunci valoarea lui `i` este 1 și valoarea lui `sum` este 1. Prin urmare, valoarea lui `sum` este 1.

Exerciții rezolvate

Exercițiul 1

Fie programul `sum` din Figura 1. Următoarele propoziții sunt adevărate sau false:

1. “Programul `sum` calculează suma primelor `n` numere naturale.”
2. “Dacă interschimbăm instrucțiunile din corpul buclei `while`, atunci `sum` nu calculează suma primelor `n` numere naturale.”
3. “Dacă valoarea de intrare pentru `n` este mai mică decât 0 atunci `suma` returnează 0.”

Răspunsuri:

Înainte de a răspunde la întrebări, vom analiza programul `suma` în detaliu. Observăm că `suma` primește la intrare un paramtru `n` de tip `int` ceea ce sugerează că inputul va fi un număr întreg. Mai departe, vom fi atenți la acest detaliu.

La început, programul `suma` inițializează variabilele `i` și `sum` cu valoarea 0. În bucla `while`, cât timp `i < n`, programul incrementează `i` și apoi adună la `sum` valoarea lui `i`. În funcție de tipul lui `n` și de condiția `i < n` avem câteva categorii de inputuri interesante:

1. `n = 0`. În acest caz, programul `suma` va returna valoarea 0, deoarece bucla nu se va executa niciodată, iar valoarea returnată este cea curentă din `sum`, adică 0.
2. `n > 0`. Vom considera pentru început că `n = 1`. În acest caz, la prima iterație, `i` devine 1, iar `sum` devine `0 + 1`, adică 1. Apoi condiția `i < n`, adică `1 < 1` nu mai este adevărată, iar programul returnează valoarea din `sum`, adică 1.

Să considerăm `n = 3`. Buclea `while` va itera de 3 ori: la prima iterație vom avea `i = 1` și `sum = 1`; la a doua iterație, `i = 2` și `sum = 3`; la a treia iterație, `i = 3` și `sum = 6`.

Observăm că (A) numărul de iterații depinde de `n`. Putem spune că programul `suma` calculează suma numerelor naturale mai mici sau egale cu `n`? Dacă tipul `int` permite stocarea unui număr întreg oricât de mare, atunci răspunsul este DA. Încercați să scrieți un program C care implementează `suma` și testați programul cu valori din ce în ce mai mari pentru `n`. Veți observa că la un moment dat programul nu mai calculează corect suma la care vă așteptați.

3. `n < 0`. În acest caz, bucla `while` nu se execută niciodată. Programul va returna 0.

Acum putem răspunde întrebărilor.

1. Răspunsul la această întrebare depinde de răspunsul la întrebarea: care sunt primele `n` numere naturale? Conform standardului ISO 80000-2:2019 (<https://www.iso.org/standard/64973.html>), 0 este considerat primul număr natural. Prin urmare, dacă pentru `n = 3`, `suma` returnează 6, iar suma primelor 3 numere naturale (0, 1 și 2) este 3, atunci răspunsul la întrebarea pusă este: *fals*.

Observați că aici a fost suficient un contraexemplu pentru a răspunde la întrebare.

2. Dacă inversăm instrucțiunile din corpul buclei `while`, atunci pentru `n = 3` vom avea: la prima iterație, `sum = 0` și `i = 1`; la a doua iterație, `sum = 1` și `i = 2`; la a treia iterație, `sum = 3` și `i = 3`. În acest caz, programul chiar calculează suma primelor `n` numere naturale, dacă `n` este pozitiv. Altfel, întrebarea nu are sens (dacă `n < 0`).

Pentru `n >= 0`, răspunsul la întrebarea pusă este: *fals*.

3. Răspuns: *adevărat* (cf. observațiilor de mai sus).

Exercițiul 4

Reamintim câteva cerințe ale Exercițiului 4 și discutăm rezolvarea acestora.

- *Coca-cola este un suc. Toate sucurile sunt lichide. Coca-cola este un lichid.*

Răspuns: raționamentul este valid. Observăm că premisele raționamentului sunt mereu adevărate: Coca-cola este într-adevăr un suc și din punct de vedere fizic, toate sucurile se află în stare lichidă. Mai mult, premisele susțin concluzia: dacă toate sucurile sunt lichide, atunci și Coca-cola, care este un tip particular de suc, este un lichid.

- *Unii točilari sunt pasionați de trenuri. Unii točilari poartă pelerine de ploaie. Deci, unii točilari pasionați de trenuri poartă pelerine de ploaie.*

Răspuns: raționamentul nu este valid și nici corect. Premizele raționamentului nu susțin concluzia. Motivul este simplu: faptul că unii točilari sunt pasionați de trenuri și unii točilari poartă pelerine nu implică faptul că există cel puțin un točilar care este în același timp pasionat de trenuri și poartă pelerină.

- *Toți studenții sunt inteligenți. Unii studenți sunt leneși. Prin urmare, unii studenți leneși sunt inteligenți.*

Răspuns: raționamentul este valid, dar nu este corect. Premizele raționamentului nu sunt mereu adevărate; spre exemplu, nu putem ști cu certitudine că toți studenții sunt inteligenți. În schimb, premisele susțin concluzia: studenții leneși fac parte din categoria (mai generală a) studenților, care, conform primei premise, sunt inteligenți.