Логічні та побітові операції

№ уроку: 5 **Курс:** Java Starter

Засоби навчання: Комп'ютер із встановленою IntelliJ IDEA

Огляд, мета та призначення уроку

Розгляд логічних функцій: кон'юнкція, диз'юнкція, заперечення, виключне АБО.
Використання побітових логічних операцій.
Розгляд логічних операцій.
Розгляд операторів зсуву.
Короткозамкнені обчислення.
Теореми де Моргана.

Вивчивши матеріал цього заняття, учень зможе:

- Розуміти роботу логічних операторів.
- Розуміти роботу бітових логічних операторів.
- Розуміти роботу операторів зсуву.
- Використовувати короткозамкнені обчислення та теореми де Моргана.

Зміст уроку

1. Розгляд логічних функцій: кон'юнкція, диз'юнкція, заперечення, виключене АБО.
2. Розгляд побітових логічних операцій.
3. Розгляд прикладу: Побітові логічні операції.
4. Розгляд прикладу: Використання побітових логічних операцій для встановлення та скидання прапорів.
5. Розгляд прикладу: Використання XOR для шифрування даних.
6. Розгляд логічних операцій (пропозиційна логіка).
7. Розгляд прикладу: Комбінація логічних операцій з операціями порівняння.
8. Розгляд прикладу: Логічний зсув вліво, вправо, а також беззнаковий зсув праворуч.
9. Розгляд прикладів: Короткозамкнуті обчислення.
10. Теореми де Моргана (Laws of dualization).
11. Розгляд прикладу: Теореми де Моргана.

Резюме

- **Кон'юнкція** (від лат. conjunctio союз, зв'язок) – логічна операція, за своїм застосуванням максимально наближена до союзу «і». Синоніми: логічне «І», логічне множення, іноді просто «І».
- **Диз'юнкція** (лат. disjunctio – роз'єднання) – логічна операція, за своїм застосуванням максимально наближена до союзу «або» в сенсі «або те, або це, або обидва відразу». Синоніми: логічне «АБО», виключне «АБО», логічне додавання, іноді просто «АБО».
- **Виключне АБО** (логічне додавання, сувора диз'юнкція) – булева функція та логічна операція. Результат виконання операції є дійсним лише за умови, якщо є дійсним у точності один з аргументів.
- **Заперечення в логіці** – унарна операція над судженнями, результатом якої є судження (у певному сенсі) «протилежне» вихідному. Позначається знаком ~ перед або над судженням. Синонім: логічне «НЕ».
- **Побітове заперечення** (або побітове НЕ, або доповнення) – це унарна операція, дія якої еквівалентна застосуванню логічного заперечення до кожного біта двійкового уявлення операнда. Іншими словами, на тій позиції, де у двійковому представленні операнда був 0, у результаті буде 1. І навпаки, де була 1, там буде 0.

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

t. 0 800 337 146
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- **Побітове І** – це бінарна операція, дія якої еквівалентна застосуванню логічного І до кожної пари бітів, які стоять на однакових позиціях у двійкових представленнях операндів. Іншими словами, якщо обидва відповідні біти операндів рівні 1, вислідний двійковий розряд дорівнює 1; якщо хоча б один біт із пари дорівнює 0, вислідний двійковий розряд дорівнює 0.
- **Побітове АБО** – це бінарна операція, дія якої еквівалентна застосуванню логічного АБО до кожної пари бітів, які стоять на однакових позиціях у двійкових представленнях операндів. Іншими словами, якщо обидва відповідні біти операндів дорівнюють 0, двійковий розряд результату дорівнює 0; якщо хоча б один біт із пари дорівнює 1, двійковий розряд результату дорівнює 1.
- **Побітове виключне АБО** (або побітове додавання за модулем два) – це бінарна операція, дія якої еквівалентна застосуванню логічного виключеного АБО до кожної пари бітів, які стоять на однакових позиціях у двійкових представленнях операндів. Інакше кажучи, якщо відповідні біти операндів різні, то двійковий розряд результату дорівнює 1; якщо біти збігаються, то двійковий розряд результату дорівнює 0.
- Бітові зрушення належать до бітових операцій. Під час зсуву значення бітів копіюються в сусідні напрямки зсуву. Розрізняють кілька різновидів зрушень: логічний, арифметичний і циклічний, залежно від обробки останніх бітів.
  Також розрізняють **зрушення вліво** (у напрямку від молодшого біта до старшого) і **вправо** (в напрямку від старшого біта до молодшого).
- Бінарні оператори & є наперед визначеними для цілих типів та bool. Для цілих типів оператор виконує бітову операцію логічного множення операндів. Для операндів boolean оператор & виконує операцію логічного множення операндів, тобто, якщо обидва операнди true, результатом буде значення true інакше false.
- Оператор & обчислює обидва операнди незалежно від першого з них.
- Для цілих типів | обчислює результат бітової операції АБО для своїх операндів. Для операндів boolean | виконує операцію логічного АБО для своїх операндів, тобто результатом буде значення false тоді й тільки тоді, коли обидва операнди мають значення false.
- Оператор ^ виконує побітову операцію виключеного OR його операндів. Для операндів boolean оператор ^ виконує операцію логічного виключеного OR операндів, тобто результатом буде значення true тільки в тому випадку, якщо рівно один із його операндів має значення true.
- **Логічний зсув**. Під час логічного зсуву значення останнього біта за напрямом зсуву втрачається (копіюючись у біт перенесення), а перший набуває нульового значення. Логічні зрушення ліворуч і праворуч використовуються для швидкого множення та поділу на 2.
- **Оператор зсуву вліво** (<<) зрушує перший операнд вліво відповідно до кількості бітів, які задані другим операндом. Тип другого операнда має бути int або тип, що має зумовлене неявне числове перетворення на int.
- Старші розряди, які розташовані не в діапазоні типу першого операнда після зміни відкидаються, а порожні молодші розряди заповнюються нулями. Оператори зсуву ніколи не викликають переповнень.
- **Оператор зсуву вправо** (>>) зсуває перший операнд вправо відповідно до кількості бітів, які задані другим операндом.
- Якщо число негативне (зі знаком -), то старші (ліворуч) розряди заповнюються одиницями, а молодші відповідно губляться. Якщо число позитивне, то старші розряди заповнюються нулями, молодші – губляться.
- Оператор зсуву вправо (>>>) зсуває перший операнд вправо відповідно до кількості бітів, які задані другим операндом. Під час використання беззнакового зсуву старші розряди **ЗАВЖДИ** заповнюються нулями незалежно від знака числа (+ або -)
- Якщо тип першого операнда int або long, зсув вправо є арифметичним зрушенням (порожнім старшим розрядам заданий знаковий біт). Якщо тип першого операнда long, початок зсуву визначається шістьма молодшими розрядами другого операнда (другий операнд & 0x3f).
- Умовний оператор AND (&&) виконує логічне AND своїх операндів типу boolean, але обчислює лише другий операнд за потреби.
- Умовний оператор OR (||) виконує логічне OR своїх операндів типу boolean, але обчислює лише другий операнд за потреби.

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

t. 0 800 337 146
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- **Короткозамкнене обчислення** – техніка, що працює за таким принципом: якщо значення першого операнда в операції AND (&&) помилкове, то другий операнд не обчислюється, бо повний вираз у будь-якому випадку буде хибним.
  Або якщо значення першого операнда в операції OR (||) є істинним, то другий операнд не обчислюється, бо повний вираз у будь-якому випадку буде істинним.
- Для застосування теорем де Моргана до логічного оператора AND або OR та пари операндів потрібно інвертувати обидва операнди, замінити (AND на OR) або (OR на AND) та інвертувати весь вираз повністю.

Закріплення матеріалу

- Назвіть основні логічні функції.
- Розкажіть таблицю істинності кон'юнкції.
- Розкажіть таблицю істинності диз'юнкції.
- Розкажіть таблицю істинності виключеного АБО.
- Де й навіщо використовуються логічні операції?
- Де й навіщо використовуються побітові логічні операції?
- Де й навіщо використовуються зсуви?
- Що таке короткозамкнені обчислення?
- Які ви знаєте короткозамкнуті обчислення?
- Дайте визначення теорем де Моргана.

Додаткове завдання

Завдання 1
Відомо, що в числах, які є ступенем двійки, лише один біт має значення 1.
Використовуючи IntelliJ IDEA, створіть клас PowerOfTwo.
Напишіть програму, яка перевірятиме, чи є вказане число ступенем двійки, чи ні.

Завдання 2
Використовуючи IntelliJ IDEA, створіть клас **DeMorganComparison**.
Використовуючи теорему де Моргана, перетворіть вихідний вираз A | B на еквівалентний вираз.

Самостійна діяльність учня

Завдання 1
Вивчіть основні конструкції та поняття, розглянуті на уроці.

Завдання 2
Відомо, що в парних числах молодший біт має значення 0. Використовуючи IntelliJ IDEA, створіть клас **Parity**. Напишіть програму, яка виконуватиме перевірку чисел на парність. Запропонуйте два варіанти розв'язку поставленого завдання.

Завдання 3
Використовуючи IntelliJ IDEA, створіть клас **LogicOperations**.
Є 3 змінні типу `int x = 5, y = 10, i z = 15;`
Виконайте та розрахуйте результат таких операцій для цих змінних:

- `x += y >> x++ * z;`
- `z = ++x & y * 5;`
- `y /= x + 5 | z;`
- `z = x++ & y * 5;`
- `x = y << x++ ^ z;`

Завдання 4

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine
t. 0 800 337 146
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

Використовуючи IntelliJ IDEA, створіть клас **Premium**. Напишіть програму розрахунку нарахування премій працівникам. Премії розраховуються згідно з вислугою років. Якщо вислуга до 5 років, премія становить 10% від заробітної плати. Якщо вислуга від 5 років (включно) до 10 років, то премія становить 15% від заробітної плати. Якщо вислуга від 10 років (включно) до 15 років, премія становить 25% від заробітної плати. Якщо вислуга від 15 років (включно) до 20 років, премія становить 35% від заробітної плати. Якщо вислуга від 20 років (включно) до 25 років, премія становить 45% від заробітної плати. Якщо вислуга від 25 років (включно) та більше, премія складає 50% від заробітної плати. Результати розрахунку виведіть на екран.

Рекомендовані ресурси

Логічні оператори
https://docs.oracle.com/javase/tutorial/java/nutsandbolts/op3.html

Усі оператори Java
https://docs.oracle.com/javase/tutorial/java/nutsandbolts/opsummary.html

CyberBionic Systematics ®
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

t. 0 800 337 146
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com