ПАВЕЛ АЗЪЛОВ • ФАНИ ЗЛАТАРОВА

C++

В ПРИМЕРИ, ЗАДАЧИ И ПРИЛОЖЕНИЯ

ПРОСВЕТА
Основано 1945 г.

ПАВЕЛ АЗЪЛОВ ФАНИ ЗЛАТАРОВА

C++ В ПРИМЕРИ, ЗАДАЧИ И ПРИЛОЖЕНИЯ

ПРОСВЕТА
София

Павел Азълов
Фани Златарова

C++ В ПРИМЕРИ, ЗАДАЧИ И ПРИЛОЖЕНИЯ

Редактори: *Милена Константинова, Невена Чардакова*
Художник на корицата и графичен дизайн *Тотко Кьосемарлиев*
Художник редактор *Вихра Стоева-Янчева*
Технически редактор *Йорданка Иванова*
Коректор *Жана Ганчева*

Българска. Издание I. Формат 70x100/16. Печ. коли 28,5. Изд. коли 37,05.
Код 20904805289.

Издателство „Просвета – София“ АД – София 1618, ул. „Земеделска“ 2
www.prosveta.bg

Печат ПК *„Димитър Благоев“* ООД – София

ISBN 978–954–01–2521–3

СЪДЪРЖАНИЕ

ПРЕДГОВОР

Настоящата книга е учебник по програмиране на езика C++ и е предназначена за широк кръг от читатели. На първо място се имат предвид учениците със засилен интерес в областта на информатиката, които изучават програмиране в рамките на профилирано обучение, а също така и студентите по информатика, които са в началните курсове на обучение. Книгата ще бъде полезна и за преподавателите, които ръководят школи по информатика. Всички използвани понятия и методи са въведени и илюстрирани с множество примери и това определя книгата като уводен курс по програмиране.

Убедени сме, че успехът в програмирането се постига с решаването на „програмистки“ задачи, т.е. задачи, за които трябва да се пишат програми. Това е процес, в който ясно се разграничават три основни стъпки: изграждане на *алгоритъм*, *проектиране* на програмата и *реализацията ѝ* на съответен език за програмиране. Тези стъпки са елементи от решенията на представените задачи.

За начинаещите програмисти е важно да се научат да „четат“ чужди програмни текстове, преди да започнат да пишат свои програми. Още преди много години Бил Гейтс (Bill Gates) казва: *„Трябва да четете програми, написани от други хора, след това да пишете свои програми, като ги предоставяте за мнение на други програмисти“*. За професионалните програмисти това е практика, която продължава и след като те вече са усвоили много от възможностите на езика и познават основните методи за програмиране.

Езикът C++ е необятен по своите възможности и приложения. За него са написани десетки компилатори и стотици книги. Въпреки големия обем на някои от тях няма учебници, които да покриват пълните му възможности. Детайли от езика се усвояват в реалната практическа работа. Това е и съветът на автора на езика, професор Бярне Строуструп (Bjarne Stroustrup), също написал много книги по C++: *„Без паника! Всичко ще стане ясно с времето. Не е нужно да знаете всеки детайл от* C++*, за да пишете добри програми. Наблегнете на техниките на програмиране, а не на езиковите особености“*.

Книгата е оформена в 15 глави и съдържа основните теми на всеки уводен курс по програмиране. Езикът C++ е обектно ориентиран език за програмиране, но в повечето от темите на тази книга са представени възможностите му, характерни за процедурно ориентираните езици за програмиране. Въпреки това идеята за обектния подход и настройката за въвеждането му започват още от началото. Разделянето на спецификацията на подпрограмите от тяхната реализация е първата стъпка в тази посока. Така всяка функция се представя на

две нива – декларация и дефиниция. По-късно тази идея прераства в отделянето на групи от функции в самостоятелни програмни модули. Паралелно с това постепено се въвеждат и започват да се използват обекти и класове, които са част от основните понятия на обектно ориентираното програмиране. Този подход е един от най-използваните в университетите у нас и по света. Това се потвърждава от съдържанието на учебниците на водещите в света издателства.

С всяка тема се въвеждат нови понятия от езика C++, илюстрирани с подходящи примери. В рамките на 64 примера и 72 решени задачи, представени с пълни програмни текстове, общият обем на които надхвърля 5300 реда програмен код на езика C++, читателят ще намери много и разнообразни идеи, които може самостоятелно да доразвие и усъвършенства. Някои от задачите се разглеждат в развитие. Всички програми са тествани в среда за програмиране Microsoft Visual Studio. Ще отбележим само, че включването на някои от библиотеките в тази среда се извършва автоматично, но в някои други среди трябва да се извършва явно.

Почти във всяка тема присъства рубриката *Приложения*. В нея са формулирани и решени програмистки задачи, чиито крайни решения са програми, написани на езика C++. Всяка програма е подробно обяснена и документирана със съответен коментар. За пълното разбиране на програмите допринасят и приведените резултати от техните експериментални изпълнения.

В края на всяка глава е включен раздел с въпроси и задачи. С въпросите се прави кратко резюме на въведените в темата понятия, а задачите са ориентирани най-вече към писане на програми. За много от задачите са дадени кратки упътвания. Последните шест теми от книгата завършват със задания на *програмни проекти*. Всеки проект представлява нетривиална задача, чието решение изисква повече време за обмисляне на алгоритъма и проектиране на програмата. В края на книгата са добавени приложения с информация, която ще бъде полезна при използването на основния текст. Към учебника са подготвени учебни материали, подпомагащи използването му както от преподаватели, така и от ученици.

Авторите изказват благодарност на рецензентите и специално на г-жа Десислава Рачева, чиито бележки бяха конструктивни и допринесоха за подобряване на ръкописа. За нас е удоволствие да изкажем благодарност и на всички сътрудници от издателство „Просвета“, с които пряко или косвено имахме възможност да работим съвместно от първоначалната идея до пълното завършване на книгата. Нашата специална благодарност е отправена към редактора на книгата, г-жа Милена Константинова, с която реализирахме този проект изцяло чрез интернет.

Авторите, август 2010

1 МАТЕМАТИКА И ПРОГРАМИРАНЕ

В продължителния и труден процес на осъзнаване на необходимостта от броене и смятане първите помощници на човека са били неговите пръсти. Природата го е дарила с този „инструмент“, като с това до голяма степен е предопределила и бройната система, в която да брои и да извършва пресмятания с числа.

Десетичната бройна система е намерила пряко отражение във всички механични устройства за смятане – в сумиращата машина на Шикард, в „колелото“ на Паскал, в аритметичната машина на Лайбниц и др.

Използването на електричество и електронни елементи в изчислителните машини, от друга страна, показва неоспоримите предимства на двоичната бройна система. Така хората се отказват от традиционната десетична система, когато решават проблеми, свързани с автоматизация на пресмятанията.

Най-съществено и най-ценното в съвременния компютър е способността му да изпълнява цялостно предварително определени системи от команди, наречени *алгоритми*. Това става възможно, след като са били намерени конструктивни решения за това, как компютърът да определя стойността на едно съждение като „истина“ или „неистина“ и по този начин сам, без външна намеса, да избира едното между две възможни действия.

От казаното дотук може да се забележи, че при създаването на съвременните компютри са използвани редица основни математически понятия и теории. Три от тях: *бройна система*, *алгебра на съжденията* и *алгоритъм*, се разглеждат по-долу.

1 БРОЙНИ СИСТЕМИ

Едно от най-древните математически открития е свързано с използването на числата. Много преди да се научат да пишат и четат, хората са изпитвали необходимост да броят. Методите за смятане с числа се развиват по-късно. Тяхната пълна автоматизация настъпва с откриването на съвременния компютър. В основата на числените пресмятания, включително и на тези, извършвани от компютри, са бройните системи.

Десетична бройна система

Обикновено човек записва и извършва пресмятания с числа в *десетична бройна система*. Азбуката на десетичните числа се състои от десет знака. Общоприето е тези знаци да се означават с 0, 1, 2, 3, 4, 5, 6, 7, 8 и 9 и да се наричат *цифри*. Числото 10 играе съществена роля в десетичната бройна система и се

нарича *основа* на бройната система. Записът на едно произволно естествено число, например 21 639, е следният:

(1) $21639 = 2 \times 10^4 + 1 \times 10^3 + 6 \times 10^2 + 3 \times 10^1 + 9 \times 10^0$.

Последователността, в която се записват отделните цифри на всяко число, е съществена. Например числото 92 631 е различно от числото 21 639, въпреки че е съставено от същите цифри. По тази причина десетичната бройна система се нарича и *позиционна*. Записването на естествените числа в десетична позиционна бройна система се основава на следната важна теорема:

Теорема. Всяко естествено число N може да се представи по единствен начин във вида:

(2) $N = a_k \times 10^k + a_{k-1} \times 10^{k-1} + ... + a_1 \times 10^1 + a_0 \times 10^0$,

където a_k, a_{k-1}, ... , a_1 и a_0 са десетични цифри и $a_k \neq 0$.

Такова представяне на числото 21 639 е дадено в (1). Без да доказваме теоремата, нека добавим, че тя е в сила и в случая, когато вместо числото 10 се използва кое да е естествено число p, $p \geq 2$.

Двоична бройна система

Десетичната бройна система не е единствената позиционна система за представяне на естествени числа. Например десетичното число 237 може да се запише в *двоична бройна система* по следния начин:

(3) $237 = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$.

За да е ясно в коя бройна система е записано едно число, в случаите, когато се налага, в скоби се добавя съответната основа. За разгледания пример имаме $237_{(10)} = 11101101_{(2)}$ и ще четем: „десетичното число 237 е равно на двоичното число 11101101“. За да се убедим във верността на това равенство, достатъчно е да пресметнем дясната страна на израза (3). Забележете, че при десетичната система се използват десет различни знака, а при двоичната бройна система те са само два (0 и 1). Когато едно число е записано в двоична бройна система, се казва, че то е *двоично число*.

Превръщане на числа от десетична в двоична бройна система и обратно

Превръщането на естествените десетични числа в двоични се извършва с неколкократно деление на числото 2.

Пример. Да превърнем десетичното число 74 в двоично число, т.е. да намерим такова двоично число $x_{(2)}$, за което

$$74_{(10)} = x_{(2)}.$$

Последователно извършваме следните операции и записваме остатъка от делението:

$74 : 2 = 37,$ т.е. $b_0 = 0$ (0 – остатък от делението на 74 на 2);

$37 : 2 = 18 + \frac{1}{2},$ т.е. $b_1 = 1$ (1 – остатък от делението на 37 на 2);

$18 : 2 = 9,$ т.е. $b_2 = 0;$

$9 : 2 = 4 + \frac{1}{2},$ т.е. $b_3 = 1;$

$4 : 2 = 2,$ т.е. $b_4 = 0;$

$2 : 2 = 1,$ т.е. $b_5 = 0;$

$1 : 2 = 0 + \frac{1}{2},$ т.е. $b_6 = 1.$

Цифрите на търсеното двоично число представляват остатъците от делението, подредени от последния към първия.

Окончателно се получава, че $74_{(10)} = 1001010_{(2)}$.

Проверка. $1001010_{(2)} = 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$
$= 64 + 8 + 2$
$= 74_{(10)}$

Направената проверка показва на практика как се превръща двоично в десетично число.

Обикновено преобразуването на естествени числа от десетична в двоична бройна система се извършва по следната по-компактна изчислителна схема (фиг. 1.1):

	: 2	: 2	: 2	: 2	: 2	: 2	: 2
цяла част	$0 \leftarrow$	$1 \leftarrow$	$2 \leftarrow$	$4 \leftarrow$	$9 \leftarrow$	$18 \leftarrow$	$37 \leftarrow 74_{(10)}$
	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$	$\downarrow$
остатък	1	0	0	1	0	1	0
	b_6	b_5	b_4	b_3	b_2	b_1	b_0

Фиг. 1.1. *Преобразуване от десетична в двоична бройна система*

В таблица 1.1 е дадено двоичното представяне на първите 16 неотрицателни числа. Всяко число е записано с четири двоични цифри.

0	0000	**4**	0100	**8**	1000	**12**	1100
1	0001	**5**	0101	**9**	1001	**13**	1101
2	0010	**6**	0110	**10**	1010	**14**	1110
3	0011	**7**	0111	**11**	1011	**15**	1111

Таблица 1.1. *Двоично представяне на числата от 0 до 15*

Други бройни системи

Разгледаните дотук десетична и двоична бройна система са частни, но важни случаи на бройните системи. За различни цели може да се използват и други бройни системи. Ще разгледаме троичната и шестнадесетичната бройна система.

Троична бройна система

Цифрите в тази система са три и се означават със знаците 0, 1 и 2. Превръщането на едно десетично число в троично се извършва по същата схема, която беше приложена за двоични числа.

Пример. Да представим десетичното число $155_{(10)}$ в троична бройна система:

$$\begin{array}{ccccccccccc} & & :3 & & :3 & & :3 & & :3 & & :3 \\ 0 & \leftarrow & 1 & \leftarrow & 5 & \leftarrow & 17 & \leftarrow & 51 & \leftarrow & 155_{(10)}, \\ & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ & & 1 & & 2 & & 2 & & 0 & & 2 \end{array}$$

т.е. $155_{(10)} = 12202_{(3)}$.

Проверка. $12202_{(3)} = 1 \times 3^4 + 2 \times 3^3 + 2 \times 3^2 + 0 \times 3^1 + 2 \times 3^0$
$= (((1 \times 3 + 2) \times 3 + 2) \times 3 + 0) \times 3 + 2$
$= ((5 \times 3 + 2) \times 3 + 0) \times 3 + 2$
$= (17 \times 3 + 0) \times 3 + 2$
$= 51 \times 3 + 2$
$= 155_{(10)}$

Шестнадесетична бройна система

Цифрите в шестнадесетичната бройна система са 16 и е прието да се означават със знаците 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E и F.

Пример. Да превърнем десетичното число $2590_{(10)}$ в шестнадесетично по вече познатата схема:

$$\begin{array}{lccccccc} & & :16 & & :16 & & :16 & \\ \text{цяла част} & 0 & \leftarrow & 10 & \leftarrow & 161 & \leftarrow & 2590_{(10)}, \\ & & \downarrow & & \downarrow & & \downarrow & \\ \text{остатък} & \text{A} & & 1 & & \text{E} & & \end{array}$$

т.е. $2590_{(10)} = \text{A1E}_{(16)}$.

2 СЪЖДИТЕЛНА ЛОГИКА

В обикновения език устно или писмено под формата на изречения изразяваме различни мисли. За някои от тях може да се постави въпросът, дали са истина, или не. Тези изречения се наричат *съждения*. В темата ще разгледаме основните понятия на т.нар. *алгебра на съжденията*, която е дял на математическата логика и е съществена част от математическите основи на компютрите.

Съждения. Верностна стойност

Изречения и съждения. Да разгледаме следните изречения:

- Числото 10 е по-голямо от 5.
- Вазов е роден в Карлово.
- Числото 25 е нечетно.
- Водородът е газ.
- Всеки ромб е квадрат.
- Земята е най-малката планета от Слънчевата система.

За всяко от тези изречения може да се зададе въпросът, дали е вярно, или не и да се отговори еднозначно с *да* или *не*. Наистина посочените примери са от разнообразни области, но не трябва да се остава с впечатлението, че всяко твърдение е съждение. Ето изречения, които не са съждения:

- Българските програмисти имат особена заслуга в компютърната вирусология. (*Вярно ли е, или не? Не може да се отговори.*)
- Това твърдение е лъжа. (*За кое твърдение става дума? Може да е лъжа, може и да не е, а възможно е изобщо да не може да се отговори.*)
- Как си ти?
- Петре, иди да купиш хляб!
- Кой е най-добрият език за програмиране?
- Ах, колко си забавна!

Определение. Изречение, изразяващо мисъл, за която може да се постави въпросът, дали е *вярна*, или *невярна*, се нарича *съждение*.

Съжденията са основни обекти на разглеждане в т.нар. алгебра на съжденията. За всяко съждение от значение е само неговата *верностна стойност*.

Верностна стойност на съжденията. Да разгледаме отново представените в началото примери. Можем еднозначно да кажем:

- Вярно е, че числото 10 е по-голямо от 5.
- Вярно е, че числото 25 е нечетно.
- Вярно е, че водородът е газ.
- Не е вярно, че Вазов е роден в Карлово.
- Не е вярно, че всеки ромб е квадрат.
- Не е вярно, че Земята е най-малката планета от Слънчевата система.

За удобство съжденията често се означават с букви, на които може да се гледа като на *съждителни променливи*. Например, да означим първите две съждения съответно с p и q:

p: Числото 10 е по-голямо от 5.

q: Вазов е роден в Карлово.

Ясно е, че променливата p има стойност истина, а променливата q е със стойност неистина. Много често думите *истина* и *вярно* се използват като синоними. Като синоними се използват и думите *неистина, невярно* и *лъжа*. Може би ще се съгласите, че още по-удобно е да използваме знаците 0 и 1 като верностни стойности на съжденията. С 1 ще означаваме стойността на съждение, което е истина, а с 0 – стойността на съждение, което е неистина.

Нека означим с x едно какво да е съждение, а неговата верностна стойност – с $v(x)$, т.е.:

$$v(x) = \begin{cases} 0, \text{ ако } x \text{ е } \textit{неистина}; \\ \\ 1, \text{ ако } x \text{ е } \textit{истина}. \end{cases}$$

Така с въведените означения за съждителните променливи p и q ще имаме $v(p) = 1$ и $v(q) = 0$.

Прости и съставни съждения

Определение. Съждение, което не съдържа в себе си друго съждение, се нарича *просто*.

Разгледаните дотук съждения бяха примери на прости съждения. *Съставните* (*сложни*) съждения се състоят от прости съждения, като например:

- Числото 10 е по-голямо от 5 и се дели на 5.
- Водородът е газ и живакът не е метал.
- Вазов е роден в Казанлък или е роден в Сопот.
- Ромбът е успоредник и няма прав ъгъл.
- Земята е планета, която е по-малка от Сатурн и е по-голяма от Меркурий.

Конюнкция, отрицание и дизюнкция

От посочените примери се вижда, че за да се образува съставно съждение от две или повече прости съждения, между тях се използват логически връзки (съюзи) като *и*, *или* и др. *Конюнкцията*, *дизюнкцията* и *отрицанието* са три основни операции в алгебрата на съжденията.

Определение. Ако едно съждение е получено от други две чрез свързването им с логическата връзка *и*, тогава се казва, че то е *конюнкция* на тези две съждения.

Ако p и q са две какви да е съждения, тяхната конюнкция се означава с $p\&q$. Приема се, че съждението $p\&q$ има стойност истина само ако и двете съжде-

ния p и q са истина. Във всички останали случаи стойността на $p\&q$ е неистина (таблица 1.2).

Примери

- Автомобилите с марка „Беемве“ са бързи *и* с лекота вдигат над 200 км/ч.
- Днес беше хубав зимен ден *и* по пистите на Витоша имаше много скиори.

Определение. Ако едно съждение е получено от други две съждения чрез свързването им с логическата връзка *или*, тогава се казва, че то е *дизюнкция* на тези две съждения.

Ако p и q са две прости съждения, тяхната дизюнкция се означава с $p \vee q$. Приема се, че съждението $p \vee q$ има стойност истина само ако поне едно от съжденията p и q е истина. Когато и двете съждения са неистина, стойността на $p \vee q$ също е неистина (таблица 1.2).

Примери

- Петър закусва с мляко или кафе, *или* и с двете.
- Ще получаваме информация за финалите на световното първенство по футбол от първи телевизионен канал или от интернет, *или* и от двата информационни източника.

Някои от съжденията са съставени чрез използване на отрицание на дадено съждение. За тази цел най-често се използват *не* или *няма*.

Определение. *Логическото отрицание* е едноаргументна операция, която, приложена над дадено съждение, променя стойността му от истина в неистина и обратно.

Примери

- Утре е първият учебен ден.
  Отрицание. Утре *не* е първият учебен ден.
- Училищният ни отбор по футбол е на първо място.
  Отрицание. Училищният ни отбор по футбол *не* е на първо място.
- Ромбът няма прав ъгъл.
  Отрицание. Ромбът *има* прав ъгъл.
- Инчът не е мярка за вместимост.
  Отрицание. Инчът *е* мярка за вместимост.

Ако p е какво да е съждение, неговото отрицание се означава с $\neg p$. Стойността на съждението $\neg p$ е истина, когато p е неистина, а когато p е истина, стойността на $\neg p$ е неистина (таблица 1.2).

$v(p)$	$v(q)$	$v(p\&q)$	$v(p \vee q)$	$v(\neg p)$	$v(\neg q)$
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	1
1	1	1	1	0	0

Таблица 1.2. *Верностни стойности на дизюнкция, конюнкция и отрицание*

Още три операции на съждителното смятане

Освен разгледаните конюнкция, дизюнкция и отрицание в алгебрата на съжденията често се използват и следните три операции: *изключваща дизюнкция*, *импликация* и *равнозначност*.

Определение. Ако p и q са две съждения, *изключващата дизюнкция* на p и q е съставно съждение, означавано с $p \oplus q$, което има верностна стойност истина, ако само едно от двете съждения p или q е със стойност истина, и е със стойност неистина в противен случай (таблица 1.3).

Примери

- Днес следобед или ще отида да гледам мач на стадиона, или ще отида да играя футбол в училище.
- Цветомир е редовен студент или в София, или в Берлин.

Често в говоримия език използваме твърдения, които най-общо записваме така: „Ако..., то...“.

Примери

- *Ако* учите редовно уроците си, *то* добрият успех ви е гарантиран.
- *Ако* в събота падне хубав сняг, *то* в неделя ще има много скиори по пистите на Витоша.

В тези примери се вижда как от две съждения p и q може да се образува ново съждение, имащо вида „Ако p, то q“. Съждение от този вид се нарича *импликация*.

Определение. *Импликацията* на две съждения p и q е съставно съждение, което се записва символично с $p \rightarrow q$ и има верностна стойност неистина само ако p е истина, а q е неистина. Във всички останали случаи верностната стойност на импликацията е истина (таблица 1.3).

Често, за да изкажем едновременно валидността и на импликацията $p \rightarrow q$, и на импликацията $q \rightarrow p$, си служим с изречения с конструкцията „p тогава и само тогава, когато q“. Такива съждения се наричат *равнозначни*. Типични примери в това отношение са формулировките на редица твърдения и теореми от математиката.

Примери

- Едно естествено число се дели на 5 *тогава и само тогава*, когато завършва на 0 или на 5.
- Диагоналите на един четириъгълник се разполовяват *тогава и само тогава*, когато четириъгълникът е успоредник.

Определение. *Равнозначността* на две съждения p и q е съставно съждение, което се записва символично с $p \leftrightarrow q$ и има верностна стойност истина,