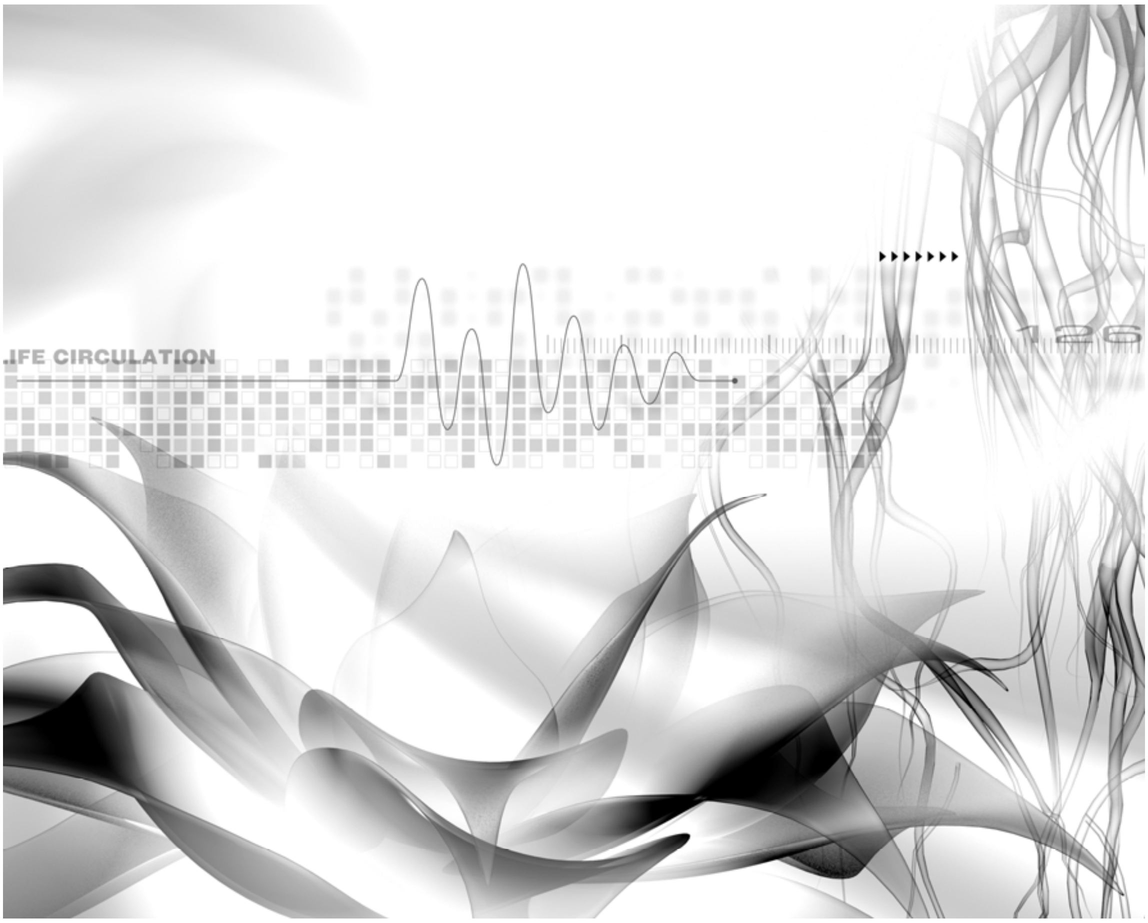
П.С. Довгий, В.И. Поляков

АРИФМЕТИЧЕСКИЕ ОСНОВЫ ЭВМ

**Учебно-методическое пособие
по выполнению домашних заданий
по дисциплине
"Дискретная математика"**



Санкт- Петербург

2010

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

ПОБЕДИТЕЛЬ КОНКУРСА ИННОВАЦИОННЫХ ОБРАЗОВАТЕЛЬНЫХ ПРОГРАММ ВУЗОВ

Довгий П.С., Поляков В.И.

АРИФМЕТИЧЕСКИЕ ОСНОВЫ ЭВМ

Учебно-методическое пособие
по выполнению домашних заданий по дисциплине
"Дискретная математика"

Санкт- Петербург

2010

Довгий П.С., Поляков В.И. Арифметические основы ЭВМ. Учебно-методическое пособие по выполнению домашних заданий по дисциплине "Дискретная математика". – СПб: СПбГУ ИТМО, 2010. – 56 с.

В пособии приводятся теоретические сведения, необходимые для выполнения домашних заданий по дисциплине «Дискретная математика» (раздел «Арифметические основы ЭВМ»). Оно содержит задание по представлению целых чисел и чисел с плавающей запятой в форматах, принятых в персональных ЭВМ и в ЭВМ общего назначения, а также задания по выполнению арифметических операций. Выполнение операций базируется на использовании методов сложения, вычитания, умножения и деления в дополнительных кодах, применяемых в современных ЭВМ. Реализация метода деления в дополнительных кодах, представленная в учебно-методическом пособии, является оригинальной и не описана в литературе.

Каждое задание, представленное в пособии, содержит свою формулировку, основные положения, знание которых необходимо студенту для его выполнения, а также большое количество примеров, поясняющих материал. В приложении приводятся контрольные вопросы и большое количество вариантов задания, что позволяет в полной мере решить проблему их индивидуализации.

Пособие предназначено для использования в учебном процессе при подготовке бакалавров и магистров по направлению 230100.68 “ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА”, а также инженеров по специальности 230101.65 “ВЫЧИСЛИТЕЛЬНЫЕ МАШИНЫ, КОМПЛЕКСЫ, СИСТЕМЫ И СЕТИ” и аспирантов.

Рекомендовано Советом факультета Компьютерных технологий и управления 15 сентября 2009 г., протокол №2

СПбГУ ИТМО стал победителем конкурса инновационных образовательных программ вузов России на 2007-2008 годы и успешно реализовал инновационную образовательную программу «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий», что позволило выйти на качественно новый уровень подготовки выпускников и удовлетворять возрастающий спрос на специалистов в информационной, оптической и других высокотехнологичных отраслях науки. Реализация этой программы создала основу формирования программы дальнейшего развития вуза до 2015 года, включая внедрение современной модели образования.

ВВЕДЕНИЕ

История систем счисления восходит к античному периоду развития математики. Высшим достижением древней арифметики является открытие позиционного принципа представления чисел. Первой из известных систем счисления, основанных на позиционном принципе, была вавилонская 60-ричная система счисления, возникшая в Древнем Вавилоне примерно во 2-м тысячелетии до новой эры.

В обыденной жизни для кодирования числовой информации используется десятичная система счисления с основанием 10. Предшественницей десятичной системы счисления является Индусская десятичная система, возникшая примерно в 8-м столетии нашей эры. Известный французский математик Лаплас выразил свое восхищение позиционным принципом и десятичной системой в следующих словах: "Мысль выражать все числа 9 знаками, придавая им, кроме значения по форме, еще значение по месту, настолько проста, что именно из-за этой простоты трудно понять, насколько она удивительна…".

В цифровых устройствах обработки информации используется двоичная система счисления с основанием 2. Современная двоичная система была полностью описана Лейбницом в XVII веке в работе «Explication de l'Arithmétique Binaire». В системе счисления Лейбница были использованы цифры 0 и 1, как и в современной двоичной системе. В 1854 английский математик Джордж Буль опубликовал знаковую работу «An investigation of the laws of thougt» ("Исследование законов мышления"), описывающую алгебраические системы применительно к логике, которая в настоящее время известна как Булева алгебра или алгебра логики. Его логическому исчислению было суждено сыграть важную роль в разработке современных цифровых электронных схем. В 1937 Клод Шеннон представил в Массачусетском технологическом институте кандидатскую диссертацию «Символический анализ релейных и переключательных схем», в которой булева алгебра и двоичная арифметика были использованы применительно к электронным реле и переключателям. На диссертации Шеннона по существу основана вся современная цифровая техника. В ноябре 1937 Джордж Штибиц создал на базе реле компьютер «Model K», который выполнял двоичное сложение. В конце 1938 Bell Labs развернула исследовательскую программу во главе со Штибицом. Созданный под его руководством компьютер, завершённый 8 января 1940, умел выполнять операции с комплексными числами. Во время демонстрации на конференции American Mathematical Society в Дармутском колледже 11 сентября 1940 Штибиц продемонстрировал возможность посылки команд удалённому калькулятору комплексных чисел по телефонной линии с использованием телетайпа. Это была первая попытка использования удалённой вычислительной машины посредством телефонной линии. Среди участников конференции были Джон фон Нейман (праотец современной архитектуры компьютеров, Джон Мокли (один из создателей первого в мире электронного компьютера ENIAC) и Норберт Винер (основоположник кибернетики и теории искусственного интеллекта), впоследствии писавшие об этом в своих мемуарах.

З А Д А Н И Е 1
ПРЕДСТАВЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ И ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ В РАЗЛИЧНЫХ ФОРМАТАХ

1. Заданное число ***A*** представить в двоично-кодированной форме:
а) в упакованном формате (BCD);
б) в неупакованном формате (ASCII).

2. Заданные число ***A*** и число ***–A*** представить в виде целого числа в форме с фиксированной запятой в формате WORD (16-битный формат).

3. Заданные числа ***A*** и ***B*** представить в форме с плавающей запятой в формате Ф1.

4. Заданные числа ***A*** и ***B*** представить в форме с плавающей запятой в формате Ф2.

5. Заданные числа ***A*** и ***B*** представить в форме с плавающей запятой в формате Ф3.

6. Найти значения чисел ***Y*** и ***Z*** по их заданным шестнадцатеричным представлениям ***R*** и ***S*** в форме с плавающей запятой в формате Ф1.

7. Найти значения чисел ***V*** и ***W*** по их заданным шестнадцатеричным представлениям ***R*** и ***S*** в форме с плавающей запятой в формате Ф2.

8. Найти значения чисел ***T*** и ***Q*** по их заданным шестнадцатеричным представлениям ***R*** и ***S*** в форме с плавающей запятой в формате Ф3.

При выполнении п.п. 3–5 задания для дробного числа B в целях увеличения точности его представления произвести симметричное округление мантиссы (округление к ближайшему).

Варианты задания приведены в табл.1 (десятичные числа ***A*** и ***B***) и в табл.2 (шестнадцатеричные числа ***R*** и ***S***) приложения 1.

1.1. ПРЕДСТАВЛЕНИЕ ЧИСЕЛ В ДВОИЧНО-КОДИРОВАННОЙ ФОРМЕ

Десятичные числа представляются в ЭВМ в двоично-кодированной форме, при этом каждая десятичная цифра кодируется с помощью четверки двоичных разрядов (двоичной тетрады):

десятичная цифра	двоичная тетрада	десятичная цифра	двоичная тетрада
0	0000	5	0101
1	0001	6	0110
2	0010	7	0111
3	0011	8	1000
4	0100	9	1001

Де-
ся-
тичные
числа
пред-
ставля-

ются с использованием либо упакованного (PACK), либо неупакованного (UNPACK) формата.

В упакованном формате в каждом байте числа кодируются две цифры, в неупакованном – одна.

Для кодирования десятичных цифр используется в основном естественный двоичный код, обычно называемый кодом “8421” (по весам разрядов двоичной тетрады).

Частным случаем неупакованного формата является код ASCII (*American Standard Code for Interchange Information*), используемый в ПК. В этом коде десятичная цифра представляется в младшей тетраде байта, а старшая тетрада принимает стандартное значение 0011.

Упакованный формат обычно называют BCD-форматом (или BCD-числом – *Binary Coded Decimal*).

Пример: ***A***=395.

а) в упакованном формате

0000 0011	1001 0101
3	9 5

б) В ASCII-формате код цифры помещается в младшую тетраду байта (в младший полубайт). Старшая тетрада байта имеет стандартное значение 0011.

0011.0011	0011.1001	0011.0101
3	9	5

1.2. ПРЕДСТАВЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ В ФОРМЕ С ФИКСИРОВАННОЙ ЗАПЯТОЙ

Особенностью представления целых чисел со знаком в форме с фиксированной запятой в современных ЭВМ является использование дополнительного кода – для отрицательных чисел.

Пример. ***A*** = 250

1.2.1. Заданное десятичное число ***A*** переводится в двоичную систему счисления:

$(250)_{10} = (11111010)_2$.

Полученное двоичное число размещается в формате таким образом, чтобы его младший разряд совпадал с крайним правым (нулевым) разрядом формата. Старшие разряды формата, включая знаковый (15-ый разряд), заполняются нулями.

В шестнадцатеричной системе счисления: $(250)_{10} = (FA)_{16}$.

$$\boxed{0}\boxed{000\ \ 0000\ \ \overbrace{1111}^{F}\ \ \overbrace{1010}^{A}}$$

15 1 0

Дополнительный код положительного числа совпадает с его прямым кодом.

1.2.2. Для представления отрицательного числа в дополнительном коде производится инвертирование цифровых разрядов прямого кода (получение обратного кода числа) с добавлением единицы в младший разряд. В знаковый разряд числа заносится единица (знак “–”).

$[-A]_{пр}$ = 1.000 0000 1111 1010 – прямой код,
$[-A]_{об}$ = 1.111 1111 0000 0101 – обратный код,
$\underline{\quad + \qquad\qquad\qquad\qquad\qquad\qquad\qquad 1}$
$[-A]_{доп}$ = 1.111 1111 0000 0110 – дополнительный код.

Для преобразования отрицательных чисел из прямого кода в дополнительный может использоваться следующее правило. Младшие нули прямого кода, включая младшую (крайнюю правую) единицу, сохраняются и в дополнительном коде, а остальные разряды прямого кода заменяются на противоположные (инвертируются) в дополнительном коде. Преобразование применяется только к цифровым разрядам числа. С использованием этого правила:

$$[-\mathrm{A}]_{доп} = 1.\underbrace{111\ 1111\ 0000\ 01}_{\text{разряды, изменяемые инвертированием}}\underbrace{10}_{\text{сохраняемые разряды}}$$

Полученное представление числа в дополнительном коде записывается в формате:

1	111 1111 0000 0110
15	1 0

1.3. ОСОБЕННОСТИ ПРЕДСТАВЛЕНИЯ ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ

Форма с плавающей запятой используется для представления так называемых действительных чисел, которые в обобщенном виде представляются следующим образом:

$$A = (-1)^{signA} \cdot M_A \cdot S^{P_A},$$

где *sign**A*** – знак числа ***A*** (0 – для положительных, 1 – для отрицательных),

$\boldsymbol{M_A}$ – мантисса числа ***A***,

S – основание порядка,

$\boldsymbol{P_A}$ – порядок числа ***A***.

Основными особенностями представления чисел с плавающей запятой в современных ЭВМ являются:

1. В качестве основания порядка в современных ЭВМ используются ***S***=2 (двоичное представление мантиссы) и ***S***=16 (шестнадцатеричное представление мантиссы). Основание 2 используется персональных

компьютерах, а 16 – в компьютерах типа Мэйн Фрейм. Порядок числа представляет собой целое число со знаком.

2. Преимущественное использование так называемых нормализованных чисел.

Число с плавающей запятой называется ***нормализованным***, если старшая цифра его мантиссы является значащей (не ноль).

3. Порядок числа представляется не как целое число со знаком в явном виде, а в виде беззнакового числа, называемого смещенным порядком или характеристикой. При этом характеристика отличается от порядка на некоторую фиксированную для данного формата величину, называемую смещением (или смещением порядка)

$$\boldsymbol{X_A = P_A + d},$$

где $\boldsymbol{X_A}$ – характеристика числа $\boldsymbol{A}$, $\boldsymbol{d}$ – смещение порядка.

Существует 2 подхода к выбору величины смещения:

1) Величина смещения равна весу старшего разряда смещенного порядка (характеристики).

2) Величина смещения равна весу старшего разряда смещения порядка, уменьшенного на 1.

Первый подход используется в основном в компьютерах типа Мэйн Фрейм, второй в ПК.

В дальнейшем формат, используемый в компьютерах типа Мэйн Фрейм будем называть ***форматом Ф1;*** формат, который раньше использовался в ЭВМ семейства VAX – ***форматом Ф2*** (представление чисел в этом формате логически понятно, и он приводится исключительно в учебных целях); формат, используемый в персональных компьютерах (стандарт IEEE-754) – ***форматом Ф3.*** Данный стандарт разработан ассоциацией IEEE (*Institute of Electrical and Electronics Engineers*) и используется для представления действительных чисел (чисел с плавающей точкой) в двоичном коде. Это самый используемый стандарт для вычислений с плавающей точкой, он используется многими микропроцессорами и логическими устройствами, а также программными средствами.

Полное название стандарта в ассоциации IEEE: *IEEE Standard for Binary Floating-Point Arithmetic* (*IEEE стандарт для двоичной арифметики с плавающей точкой*. В августе 2008 года ассоциация IEEE выпустила стандарт *ANSI/IEEE Std 754-2008* (*ANSI: American National Standards Institute*– Американский национальный институт стандартов).

4. Формат чисел может быть коротким (32 бита), длинным (64 бита) и расширенным (128 бит). В домашнем задании используется только короткий (32 бита) формат.

1.4. ПРЕДСТАВЛЕНИЕ ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ В ФОРМАТЕ Ф1

Для представления числа в форме с плавающей запятой определяются его мантисса и порядок. В связи с тем, что в формате Ф1 в качестве основания порядка используется число 16, для представления мантиссы удобнее использовать исходное число в шестнадцатеричной системе счисления. Для определения мантиссы и порядка производится перемещение запятой в шестнадцатеричном числе влево или вправо таким образом, чтобы она установилась перед старшей значащей цифрой. Полученное дробное число представляет мантиссу числа с плавающей запятой. Модуль порядка определяется количеством шестнадцатеричных цифр, на которое была перенесена запятая. Знак порядка определяется направлением, в котором переносилась запятая. При перенесении запятой влево порядок положителен, вправо – отрицателен.

Такой способ получения мантиссы и порядка дает нормализованное число, у которого старшая цифра мантиссы значащая.

Если в исходном числе запятая находится перед старшей значащей цифрой, то порядок этого числа равен нулю.

По значению порядка определяется характеристика (смещенный порядок) числа путем сложения порядка со смещением (для формата Ф1 величина смещения равна 64), после чего двоичные значения знака, характеристики и мантиссы числа записываются в формат.

1.4.1. Представим в формате Ф1 число ***A***=250.

Для определения мантиссы и порядка числа ***A*** в его шестнадцатеричном представлении запятая переносится влево на две шестнадцатеричные цифры:

$$A = (\mathtt{FA})_{16} = \underbrace{(\mathtt{0,FA})_{16}}_{\text{мантисса}} \times \mathtt{16}^{\underset{\text{порядок}}{\uparrow 2}}$$

Характеристика числа ***A***: $X_A = P_A + 64 = (66)_{10} = (1000010)_2$.

Для получения двоичного значения характеристики, соответствующей положительному или нулевому порядку, в ее старший разряд записывается единица (вес этого разряда характеристики равен величине смещения – 64), а в младших разрядах характеристики представляется величина порядка

$$X_A = (\mathtt{1000010}).$$

$$\underbrace{64}_{\text{смещение}} + \underbrace{2}_{\text{порядок}} = \underbrace{66}_{\text{характеристика}}$$

При записи числа в формате шестнадцатеричная мантисса представляется в двоичной системе счисления. Младшие разряды мантиссы заполняются нулями. Представление числа ***A*** в формате Ф1 имеет вид:

		F	A	0	0	0	0
0	100 0010	1111	1010	0000	0000	0000	0000
0	1 7	8					31
знак “+”	характеристика	мантисса					

Старшая тетрада мантиссы нормализованного числа может содержать от одного до трех старших нулей. Например, число

$(1E8,5)_{16} = (0,1E85)_{16} \times 16^{3}$ представляется в виде

	64	3	1	E	8	5	0	0
0	100	0011	0001	1110	1000	0101	0000	0000
0	1	7	8					31

и содержит в старшей тетраде мантиссы три нуля.

1.4.2. Представим в формате Ф1 число ***B*** = 0,0025. Переведем число ***B*** в шестнадцатеричную систему счисления. При переводе необходимо получить шесть цифр, не считая старших нулей.

В целях повышения точности представления числа рекомендуется получить еще одну (дополнительную) цифру, по значению которой производится симметричное округление этого числа.

$\boldsymbol{B} = (0{,}0025)_{10} = (0{,}00A3D70A3)_{16}$.

Дополнительная цифра числа, равная $(3)_{16} = (0011)_2$, содержит в старшем двоичном разряде ноль и поэтому не изменяет значения предыдущей цифры, равной $(A)_{16}$, при округлении числа.

При наличии старшей единицы в двоичном представлении дополнительной цифры, что соответствует значению, большему или равному $(8)_{16}$, *при симметричном округлении предыдущая младшая цифра числа увеличивается на единицу.*

Для определения мантиссы и порядка числа ***B*** запятая в его шестнадцатеричном представлении переносится вправо на две цифры, что определяет порядок числа, равный (–2):

$$\boldsymbol{B} = (0{,}00A3D70A)_{16} = (0{,}A3D70A)_{16} \times 16^{-2}.$$

Характеристика числа ***B***: $X_B = P_B + 64 = -2 + 64 = 62 = (0111110)_2$.

Для получения двоичного значения характеристики, соответствующей отрицательному порядку, можно использовать следующее правило: в старший разряд характеристики записывается ноль, а в шести ее младших разрядах представляется дополнительный код порядка (дополнение до 64)

000010 – *прямой код порядка,*

111110 – *дополнительный код порядка,*

0111110 – *характеристика.*

Представление числа ***B*** в формате Ф1 имеет вид:

		A	3	D	7	0	A
0	011 1110	1010	0011	1101	0111	0000	1010
0	1 … 7	8					31

1.5. ПРЕДСТАВЛЕНИЕ ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ В ФОРМАТЕ Ф2

Для представления чисел в форме с плавающей запятой в формате Ф2 используется их двоичная запись, так как в этом формате основание порядка $\boldsymbol{S} = 2$. Для определения мантиссы и порядка запятая переносится влево или вправо в двоичном числе до установления перед старшей единицей. Модуль порядка определяется количеством двоичных цифр (разрядов), на которое переносится запятая.

Характеристика (смещенный порядок) определяется путем сложения порядка со смещением, величина которого в формате Ф2 равна 128 (в отличие от формата Ф1 в формате Ф2 под характеристику отводится 8 двоичных разрядов).

При записи числа необходимо учитывать, что в формате Ф2 используются **только** нормализованные числа и, так как нормализация осуществляется с точностью до двоичной цифры, старший разряд мантиссы всегда равен единице, в связи с чем он в разрядной сетке не представляется (так называемый скрытый разряд). Кроме того, в формате Ф2 принята нумерация разрядов в сетке справа налево (от младшего разряда к старшему).

1.5.1. Определение мантиссы и порядка числа $\boldsymbol{A}$:

$$\boldsymbol{A} = (250)_{10} = (FA)_{16} = (11111010)_2 = \underbrace{(0,1111101)_2}_{\text{мантисса}} \times 2^{\overset{\text{порядок}}{8}}.$$

Характеристика числа $\boldsymbol{A}$:

$$X_A = P_A + 128 = 136 = (10001000)_2.$$

$$\underbrace{128}_{\text{смещение}} + \underbrace{8}_{\text{порядок}} = \underbrace{136}_{\text{характеристика}}$$

Число $\boldsymbol{A}$ представляется в формате Ф2 в виде

0	10001000	11110100000000000000000
31	30 … 23	22 … 0
знак	характеристика	мантисса

1.5.2. Для определения мантиссы и порядка числа $\boldsymbol{B}$ запятая в его двоичном представлении переносится вправо на 8 двоичных разрядов, что определяет порядок числа, равный (–8).

$$B = (0{,}0025)_{10} = (0{,}00A3D70A)_{16} =$$

$$= (0{,}\underbrace{0000}_{0}\ \underbrace{0000}_{0}\ \underbrace{1010}_{A}\ \underbrace{0011}_{3}\ \underbrace{1101}_{D}\ \underbrace{0111}_{7}\ \underbrace{0000}_{0}\ \underbrace{1010}_{A})_2 =$$

$$= \underbrace{(0{,}10100011110101110000101)_2}_{\text{мантисса}} \times 2^{\overset{\uparrow}{-8}}.$$

порядок

Характеристика числа $\boldsymbol{B}$: $X_B = P_B + 128 = 120 = (\underbrace{01111000}_{\text{дополнительный код порядка}})_2$.

Представление числа $\boldsymbol{B}$ в формате Ф2 имеет вид:

знак	характеристика	мантисса
0	01111000	01000111101011100001010
31	30 … 23	22 … 0

Для рассмотренных примеров дробное число ***B*** *представлено в форматах Ф1 и Ф2 с одинаковой точностью. Это объясняется наличием единицы в старшем разряде первой тетрады мантиссы формата Ф1. Если же в старшем разряде первой тетрады мантиссы ноль (первая шестнадцатеричная цифра меньше 8), то точность представления числа в формате Ф2 может быть больше за счет использования большего числа значащих цифр в мантиссе.*

Например, шестнадцатеричная мантисса $(0{,}1A90BC7)_{16}$ будет представлена в формате Ф1 в виде

$$\overbrace{0001}^{1}\ \overbrace{1010}^{A}\ \overbrace{1001}^{9}\ \overbrace{0000}^{0}\ \overbrace{1011}^{B}\ \overbrace{1100}^{C}\,|\,0$$

8 … 31

разряд, по которому производится округление

а в формате Ф2 - в виде

$$1\,|\,10101001000010111100100\,|\,1$$

22 … 0

скрытый разряд

разряд, по которому производится округление

В данном случае в формате Ф2 в представлении мантиссы используется на три разряда больше, чем в Ф1. Кроме того, за счет дополнительного разряда мантиссы, равного единице, производится добавление единицы к младшему разряду, в результате чего получается представление числа с избытком, в то время как в формате Ф1 оно представлено с недостатком.

1.6. ПРЕДСТАВЛЕНИЕ ЧИСЕЛ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ В ФОРМАТЕ Ф3

Представление чисел в формате Ф3 во многом аналогично их представлению в формате Ф2. Основными отличиями являются:

1) величина смещения равна 127 (в формате Ф2 – 128);

2) старшая единица мантиссы нормализованного числа является единицей целой части мантиссы, т.е. запятая в мантиссе фиксируется после старшей единицы (в формате Ф2 запятая в мантиссе фиксируется перед старшей единицей).

1.6.1. Определение мантиссы и порядка числа ***A***:

$$\boldsymbol{A} = (250)_{10} = (\mathrm{FA})_{16} = (11111010)_2 = \underbrace{(1{,}111101)_2}_{\text{мантисса}} \times 2^{\overset{\text{порядок}}{7}}.$$

Смещенный порядок числа ***A***:

$X_A = P_A + 127 = 134 = (10000110)_2$.

Число ***A*** представляется в формате Ф3 в виде

0	1 0 0 0 0 1 1 0	1 1 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
31	**30 … 23**	**22 … 0**
знак	смещенный порядок	мантисса

Следует отметить, что:

а) в отличие от представления чисел в форматах Ф1 и Ф2 в Ф3 не принято называть смещенный порядок характеристикой;

б) по аналогии с представлением чисел в формате Ф2 в Ф3 используется скрытый разряд (единица целой части мантиссы в формате не представляется);

в) представление числа в формате Ф3 отличается от представления в Ф2 только значением смещенного порядка (его величина уменьшается на 2).

1.6.2. Определение мантиссы и порядка числа ***B***:

$$\boldsymbol{B} = (0{,}0025)_{10} = (0{,}00\mathrm{A}3\mathrm{D}70\mathrm{A})_{16} =$$

$$= (0{,}\underbrace{0000}_{0}\ \underbrace{0000}_{0}\ \underbrace{1010}_{\mathrm{A}}\ \underbrace{0011}_{3}\ \underbrace{1101}_{\mathrm{D}}\ \underbrace{0111}_{7}\ \underbrace{0000}_{0}\ \underbrace{1010}_{\mathrm{A}})_2 =$$

$$= \underbrace{(1{,}0100011110101110000101)_2}_{\text{мантисса}} \times 2^{\overset{\text{порядок}}{-9}}.$$

Смещенный порядок числа ***B***:

$X_B = P_B + 127 = 118 = (01110110)_2$.

Для чисел с отрицательным порядком значение смещенного порядка может быть получено по следующему правилу: старший разряд смещенного порядка равен нулю, а в остальных разрядах представляется обратный код порядка:

0001001 *– прямой код порядка,*
1110110 *– обратный код порядка,*
01110110 *– смещенный порядок.*

Представление числа ***B*** в формате Ф3 имеет вид:

0	01110110	01000111101011100001010
31	30 … 23	22 … 0

1.7. ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ ЧИСЛА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ ПО ЕГО ПРЕДСТАВЛЕНИЮ В ФОРМАТЕ Ф1

R = C318FC00, ***S*** = 3E600000

1.6.1. Для определения значения числа ***Y*** производится наложение его шестнадцатеричного представления ***R*** на разрядную сетку формата Ф1:

	C	3	1	8	F	C	0	0
1	100	0011	0001	1000	1111	1100	0000	0000
0	1	7	8					31
знак	характеристика		мантисса					

Из этого представления видно, что число ***Y*** – отрицательное (в знаковом разряде числа – единица).

Определим порядок числа ***Y*** по его характеристике:

$X_Y = 67 = \underbrace{64}_{\text{смещение}} + \underbrace{3}_{\text{порядок}},$

$P_Y = X_Y - 64 = 3.$

Представим число ***Y*** с помощью мантиссы и порядка:

$\boldsymbol{Y} = -\,(0{,}18FC)_{16} \times 16^3.$

Получили представление числа ***Y*** в нормальной (полулогарифмической) форме. Для приведения числа ***Y*** к естественной форме необходимо перенести запятую в мантиссе на количество шестнадцатеричных цифр, равное модулю порядка, вправо – при положительном или влево – при отрицательном порядке. В данном случае запятая переносится вправо:

$\boldsymbol{Y} = -\,(18F{,}C)_{16}.$

Переведем число ***Y*** из шестнадцатеричной в десятичную систему счисления с использованием весов разрядов:

$\boldsymbol{Y} = -\,(1 \times 16^2 + 8 \times 16^1 + 15 \times 16^0 + 12 \times 16^{-1}) =$
$= -\,(256 + 128 + 15 + 0{,}75) = -399{,}75.$

1.7.2. Для определения значения числа $\boldsymbol{Z}$ производится наложение его шестнадцатеричного представления $\boldsymbol{S}$ на разрядную сетку:

```
   3     E      6     0     0     0     0     0
|0|011  1110|0110  0000  0000  0000  0000  0000|
 0 1       7 8                                31
```

Порядок числа $\boldsymbol{Z}$:

$P_Z = X_Z - 64 = 62 - 64 = -2.$

Значение числа $\boldsymbol{Z}$:

$\boldsymbol{Z} = (0{,}6)_{16} \times 16^{-2} = (0{,}006)_{16} = 6/16^3 = 6/2^{12} = 3/2^{11} =$

$= (3/2) \times (1/2^{10}) = (3/2) \times (1/1024) \approx 1{,}5 \times 10^{-3}.$

При переводе дробных чисел из двоичной системы счисления в десятичную можно считать: $2^{-10} \approx 10^{-3}$.

1.8. ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ ЧИСЛА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ ПО ЕГО ПРЕДСТАВЛЕНИЮ В ФОРМАТЕ Ф2

1.8.1. Представление числа $\boldsymbol{V}$ в формате Ф2 имеет вид:

```
   C    3    1     8     F     C     0     0
|1|10000110|001  1000  1111  1100  0000  0000|
31 30     23 22                              0
знак  характеристика          мантисса
```

Порядок числа $\boldsymbol{V}$:

$P_V = X_V - 128 = 134 - 128 = 6.$

Значение числа $\boldsymbol{V}$ в нормальной форме:

$\boldsymbol{V} = -(0{,}\underbrace{10011000111111}_{\text{мантисса}})_2 \times 2^{6}.$

скрытый разряд   мантисса   порядок

При определении двоичного значения мантиссы производится восстановление ее скрытого старшего разряда, равного 1.

Для приведения числа $\boldsymbol{V}$ к естественной форме запятая в его мантиссе переносятся вправо на 6 двоичных разрядов:

$\boldsymbol{V} = -(100110{,}00111111)_2.$

Перевод числа $\boldsymbol{V}$ из двоичной системы в десятичную:

а) целая часть:

$(100110)_2 = 2^5 + 2^2 + 2^1 = 32 + 4 + 2 = 38;$

б) дробная часть:

первый способ перевода:

$(0{,}00111111)_2 = 2^{-3} + 2^{-4} + 2^{-5} + 2^{-6} + 2^{-7} + 2^{-8} =$

$= 1/8 + 1/16 + 1/32 + 1/64 + 1/128 + 1/256 = 63/256 \approx 0{,}246;$

второй способ перевода:

$(0{,}00111111)_2 = (111111)_2 \times 2^{-8} = 63 \times (1/256) \approx 0{,}246;$

третий способ перевода:

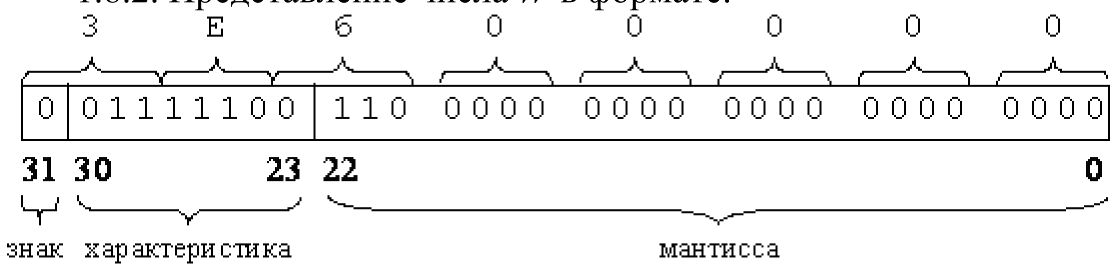
$(0{,}0011111)_2 = (0{,}01)_2 - (0{,}00000001)_2 = 1/4 - 1/256 \approx$

$\approx 0{,}25 - 0{,}004 = 0{,}246.$

Значение числа ***V***:

$\boldsymbol{V} \approx -38{,}246.$

1.8.2. Представление числа ***W*** в формате:



Порядок числа ***W***:

$P_W = X_W - 128 = 124 - 128 = -4.$

Число ***W*** в нормальной форме:

$\boldsymbol{W} = (0{,}111)_2 \times 2^{-4}.$

Число ***W*** в естественной форме получается переносом запятой в мантиссе влево на четыре двоичных разряда:

$\boldsymbol{W} = (0{,}0000111)_2.$

Значение числа ***W***:

$\boldsymbol{W} = (0{,}0000111)_2 = (111)_2 \times 2^{-7} = 7 / 128 \approx 0{,}0547.$

1.9. ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ ЧИСЛА С ПЛАВАЮЩЕЙ ЗАПЯТОЙ ПО ЕГО ПРЕДСТАВЛЕНИЮ В ФОРМАТЕ Ф3

1.9.1. Представление числа ***T*** в формате Ф3 имеет тот же вид, что и для числа ***V*** в формате Ф2 (п. 1.8.1).

Порядок числа ***T***:

$P_T = X_T - 127 = 134 - 127 = 7.$

Значение числа ***T*** в двоичной системе счисления:

$$T = -(\underbrace{1,0011000111111}_{\text{мантисса}})_2 \times 2^{7}.$$

скрытый разряд — мантисса — порядок

Перевод числа ***T*** из двоичной системы счисления в десятичную:

а) целая часть:

$(10011000)_2 = 2^7 + 2^4 + 2^3 = 128 + 16 + 8 = 152.$

б) дробная часть:

$(0{,}111111)_2 = 1 - (0{,}000001)_2 = 1 - 1 / 64 \approx 0{,}984.$

Значение числа ***T***:

$\boldsymbol{T} \approx -152{,}984.$

*По сравнению со значением числа **V**, имеющего то же самое представление в формате, число **T** в четыре раза больше за счет большего на единицу значения порядка и за счет использования скрытой целой единицы в мантиссе (мантисса представляется неправильной дробью).*

1.9.2. Представление числа $\boldsymbol{Q}$ в формате Ф3 имеет тот же вид, что и для числа в Ф2 (п.1.8.2.).

Порядок числа $\boldsymbol{Q}$:

$P_Q = X_Q - 127 = 124 - 127 = -3.$

Значение числа $\boldsymbol{Q}$:

$\boldsymbol{Q} = (1{,}11)_2 \times 2^{-3} = (0{,}00111)_2 = (111)_2 \times 2^{-5} = 7 / 32 \approx 0{,}219.$

ЗАДАНИЕ 2
СЛОЖЕНИЕ ЦЕЛЫХ ЧИСЕЛ

Сложение целых чисел выполняется в байтном формате.

1. Для заданных чисел ***A*** и ***B*** выполнить операцию знакового сложения со всеми комбинациями знаков операндов (4 примера).

Для каждого примера:

а) Проставить межразрядные переносы, возникающие при сложении.

б) Дать знаковую интерпретацию операндов и результата. При получении отрицательного результата предварительно преобразовать его из дополнительного кода в прямой.

в) Дать беззнаковую интерпретацию операндов и результата. При получении неверного результата пояснить причину его возникновения.

г) Показать значения арифметических флагов для каждого из примеров.

2. Сохранив значение первого операнда ***A***, выбрать такое значение операнда ***B***, чтобы в операции сложения с одинаковыми знаками имел место особый случай переполнения формата. Выполнить два примера, иллюстрирующие эти случаи. Для каждого из них – то же, что в п.1.

3. Сохранив операнд ***B***, подобрать такое значение операнда ***A***, чтобы при сложении положительных операндов имело место переполнение формата, а при сложении отрицательных операндов (таких же по модулю, как и положительных) результат операции был бы корректным. Выполнить два примера, для которых сделать то же, что и в п.1 и дать необходимые пояснения.

Основные положения. При сложении целых чисел с фиксированной запятой с использованием дополнительных кодов знаковый разряд участвует в операции точно так же, как и любой цифровой. При этом знак суммы формируется автоматически путем сложения знаковых разрядов операндов с учетом переноса из старшего цифрового разряда. Перенос из знакового разряда не участвует в формировании результата. Результат операции получается в дополнительном коде.

Арифметические флаги формируются в основном арифметическими командами, определяя результат арифметических операций (точнее, они являются признаками результата). Кроме того, на арифметические флаги оказывают влияние логические команды и команды сдвигов. К арифметическим флагам относятся:

CF – Carry Flag (флаг переноса). В нем фиксируется перенос из старшего разряда при сложении и заем в старший разряд при вычитании. При умножении значение флага ***CF*** определяет возможность (***CF***=0) или невозможность (***CF***=1) представления результата (произведения) в том же формате, что и сомножители.

С помощью флага переноса фиксируется переполнение при сложении беззнаковых чисел.

PF – Parity Flag – флаг паритета (четности). Он устанавливается при наличии четного числа единиц в младшем байте результата, в противном случае – сбрасывается. Этот флаг используется как аппаратная поддержка контроля по четности (нечетности).

AF – Auxiliary Carry Flag (флаг вспомогательного переноса). В этом флаге фиксируется межтетрадный перенос при сложении и межтетрадный заем при вычитании. Значение этого флага используется командами десятичной и ASCII-коррекции сложения и вычитания.

ZF – Zero Flag (флаг нуля). Он устанавливается при нулевом результате операции, в противном случае - сбрасывается.

SF – Sign Flag (флаг знака). В него копируется старший (крайний левый) бит результата, интерпретируемый как знак.

OF – Overflow Flag (флаг переполнения). Он устанавливается в командах сложения и вычитания в случае, если результат операции не помещается в формате операндов. При этом как операнды, так и результат интерпретируются как знаковые целые числа. В аппаратную установку этого флага положен принцип фиксации переполнения по сравнению переносов из двух старших разрядов при сложении или заемов в два старших разряда при вычитании. Если один из переносов (заемов) имеет место, а другой отсутствует, то происходит переполнение формата (разрядной сетки).

Варианты заданий приведены в табл. 3 приложения 1.

ПРИМЕР ВЫПОЛНЕНИЯ ЗАДАНИЯ

Операция двоичного сложения реализуется поразрядно, начиная с младших разрядов с учетом возникающих межразрядных переносов. В каждом разряде сложение реализуется в соответствии со следующей таблицей:

a_i – значение i –го разряда 1-го слагаемого,
b_i – значение i –го разряда 2-го слагаемого,
p_{i-1} –перенос из (i-1)-го разряда в i –й разряд,
S_i – сумма i-го разряда,
p_i – перенос из i –го разряда в (i+1)-й разряд

a_i	b_i	p_{i-1}	s_i	p_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Напомним, что диапазон представления чисел в байтном формате выглядит следующим образом:

- целых знаковых: $\underbrace{-128}_{\boxed{1.0000000}} =- 2^7 \le A_Ц^{ЗН} \le 2^7 - 1=\underbrace{127}_{\boxed{0.1111111}}$

- целых беззнаковых: $0 \le A_Ц^{БЗН} \le 2^8 - 1 = 255$

2.1. $\boldsymbol{A} = 57$, $\boldsymbol{B} = 49$

	7	6 … 0			7	6 … 0
$\boldsymbol{A}$	0	111001		$\boldsymbol{B}$	0	110001
$-\boldsymbol{A}$	1	000111		$-\boldsymbol{B}$	1	001111

1) $\boldsymbol{A}>0$, $\boldsymbol{B}>0$

		Знаковая интерпретация **ЗИ**	Беззнаковая интерпретация **БзИ**
$\boldsymbol{A}_{ПК}$ =	0.0 1 1 1 0 0 1	57	57
$\boldsymbol{B}_{ПК}$ = +	0.0 1 1 0 0 0 1	+ 49	+ 49
$\boldsymbol{C}_{ПК}$ =	0.1 1 0 1 0 1 0	106	106

Установка флагов:

CF = 0 - флаг переноса (отсутствие переноса из старшего разряда);

SF = 0 - флаг знака (положительный результата);

ZF = 0 - флаг нуля (отсутствие нулевого результата);

AF = 0 - флаг дополнительного переноса (отсутствие межтетрадного переноса);

PF = 1 - флаг паритета (число единиц четно);

OF = 0 - флаг переполнения (переполнение для знаковых чисел отсутствует).

2) $\boldsymbol{A}<0$, $\boldsymbol{B}>0$

		ЗИ	**БзИ**
$\boldsymbol{A}_{ДК}$=	1.1 0 0 0 1 1 1	–57	199
$\boldsymbol{B}_{ПК}$ = +	0.0 1 1 0 0 0 1	+ 49	+ 49
$\boldsymbol{C}_{ДК}$ =	1.1 1 1 1 0 0 0		248
$\boldsymbol{C}_{ПК}$ =	1.0 0 0 1 0 0 0	–8	

CF = 0; SF = 1; ZF = 0; AF = 0; PF = 0; OF = 0.

3) $\boldsymbol{A}>0$, $\boldsymbol{B}<0$

		ЗИ	**БзИ**
$\boldsymbol{A}_{ПК}$ =	0.0 1 1 1 0 0 1	57	57
$\boldsymbol{B}_{ДК}$ = +	1.1 0 0 1 1 1 1	+ –49	+ 207
$\boldsymbol{C}_{ПК}$ =	0.0 0 0 1 0 0 0	8	8?

CF=1; SF=0; ZF=0; AF=1; PF=0; OF=0.

Для БзИ результат неверен вследствие возникающего переноса из старшего разряда. Вес этого переноса составляет 256 (8+256 = 264 = =57+207).

4) $\boldsymbol{A}<0$, $\boldsymbol{B}<0$

		ЗИ	**БзИ**
$\boldsymbol{A}_{ДК}$ =	1.1 0 0 0 1 1 1	–57	199
$\boldsymbol{B}_{ДК}$ = +	1.1 0 0 1 1 1 1	+ –49	+ 207
$\boldsymbol{C}_{ДК}$ =	1.0 0 1 0 1 1 0		150 ?
$\boldsymbol{C}_{ПК}$ =	1.1 1 0 1 0 1 0	–106	

CF=1;   SF=1;   ZF=0;   AF=1;   PF=1;   OF=0.

Для БзИ результат неверен вследствие возникающего переноса из старшего разряда. Вес этого переноса составляет 256 (150+256 = = 406 = 199+207).

2.2. Правило для подбора операнда ***B*** выглядит следующим образом:

$$\boldsymbol{A} + \boldsymbol{B} > 128, \text{ значит } \quad 128 - \boldsymbol{A} < \boldsymbol{B} \leq 127.$$

A = 57 (неизменно), ***B*** = 96 (подбираем)

	7	6 … 0
+ ***B***	0	110000
- ***B***	1	010000

1) ***A*** > 0, ***B*** > 0

		ЗИ	БзИ
$A_{ПК}$=	0.0 1 1 1 0 0 1	57	57
$B_{ПК}$=	+ 0.1 1 0 0 0 0 0	+ 96	+ 96
$C_{ДК}$=	1.0 0 1 1 0 0 1		153
$C_{ПК}$=	1.1 1 0 0 1 1 1	-103?	

CF = 0;   SF = 1;   ZF = 0;   AF = 0;   PF = 1;   OF = 1.

Для знакового сложения результат является некорректным вследствие переполнения формата.

О наличии переполнения можно судить по двум фактам:

- Сравнение знаков операндов и результата. Если знаки операндов одинаковы, а знак суммы отличается от них – фиксируется переполнение.
- Сравнение переносов из старшего цифрового разряда в знаковый и из знакового разряда за пределы формата. Переполнение возникает, если один из переносов есть, а второго нет.

2) ***A*** < 0, ***B*** < 0

		ЗИ	БзИ
$A_{ДК}$=	1.1 0 0 0 1 1 1	-57	199
$B_{ДК}$=	+ 1.0 1 0 0 0 0 0	+ -96	+ 160
$C_{ПК}$=	0.1 1 0 0 1 1 1	+103?	103?

CF = 1;   SF = 0;   ZF = 0;   AF = 0;   PF = 0;   OF = 1.

Для ЗИ результат неверен вследствие возникающего переполнения (см. выше), для БзИ результат неверен вследствие возникающего переноса из старшего разряда.

2.3. Значение числа ***B*** фиксируем (***B*** = 49), а значение ***A*** определяется в соответствии с условием ***A*** + ***B*** = 128, по которой при сложении положительных чисел будет фиксироваться переполнение, а при сложении отрицательных - не будет. ***A*** = 79.

	7	6 … 0
+ ***A***	0	1001111
- ***A***	1	0110001

1) $\boldsymbol{A} > 0$, $\boldsymbol{B} > 0$

$$
\begin{array}{llll}
 & & \textbf{ЗИ} & \textbf{БзИ} \\
\boldsymbol{A}_{\text{ПК}} = & 0.1001111 & 79 & 79 \\
\boldsymbol{B}_{\text{ПК}} = & {}^{+}\underline{0.0110001} & {}^{+}\underline{49} & {}^{+}\underline{49} \\
\boldsymbol{C}_{\text{ДК}} = & 1.0000000 & -128? & 128
\end{array}
$$

CF = 0; SF = 1; ZF = 0; AF = 1; PF = 0; OF = 1.

Для ЗИ результат неверен вследствие возникающего переполнения (см. выше).

2) $\boldsymbol{A} < 0$, $\boldsymbol{B} < 0$

$$
\begin{array}{llll}
 & & \textbf{ЗИ} & \textbf{БзИ} \\
\boldsymbol{A}_{\text{ДК}} = & 1.0110001 & -79 & 177 \\
\boldsymbol{B}_{\text{ДК}} = & {}^{+}\underline{1.1001111} & {}^{+}\underline{-49} & {}^{+}\underline{207} \\
\boldsymbol{C}_{\text{ПК}} = & 1.0000000 & -128 & 128?
\end{array}
$$

CF = 1; SF = 1; ZF = 0; AF = 1; PF = 0; OF = 0.

Для БзИ результат неверен вследствие возникающего переноса из старшего разряда, результат ЗИ корректен.

З А Д А Н И Е 3
ВЫЧИТАНИЕ ЦЕЛЫХ ЧИСЕЛ

Вычитание целых знаковых чисел производится в байтном формате.

Выполнить те же пункты, что и в задании 2, применительно к операции вычитания, в пунктах 2 и 3 переполнение имеет место при разных знаках операндов.

Варианты заданий приведены в табл.3 приложения 1.

ПРИМЕР ВЫПОЛНЕНИЯ ЗАДАНИЯ

Операция двоичного вычитания реализуется поразрядно, начиная с младших разрядов с учетом возникающих межразрядных заемов.

В каждом разряде вычитание реализуется в соответствии со следующей таблицей:

a_i – значение разряда уменьшаемого,

b_i – значение разряда вычитаемого,

z_{i-1} – значение заема из i-го разряда в (i-1)-й предыдущий младший разряд,

r_i – значение разности в i-ом разряде,
z_i – значение заема из предыдущего (i+1)-го старшего разряда в i-й разряд.

a_i	b_i	z_{i-1}	r_i	z_i
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

3.1. $A = 67$, $B = 51$

$A = (67)_{10} = (1000011)_2$

$B = (51)_{10} = (110011)_2$

1) $A > 0$, $B > 0$

		ЗИ	**БзИ**
$A_{ПК=}$	_ 0.1 0 0 0 0 1 1	_ 67	_ 67
$B_{ПК=}$	0.0 1 1 0 0 1 1	51	51
$C_{ПК=}$	0.0 0 1 0 0 0 0	16	16

CF = 0; SF = 0; ZF = 0; AF = 0; PF = 0; OF = 0.

2) $A < 0$, $B > 0$

		ЗИ	**БзИ**
$A_{ДК=}$	_ 1.0 1 1 1 1 1 0 1	_- 67	_189
$B_{ПК=}$	0.0 1 1 0 0 1 1	+51	51
$C_{ДК=}$	1.0 0 0 1 0 1 0		138
$C_{ПК=}$	1.1 1 1 0 1 1 0	- 118	

CF = 0; SF = 1; ZF = 0; AF = 0; PF = 0; OF = 0.

3) $A > 0$, $B < 0$

		ЗИ	БзИ
$A_{ПК}=$	_ 0.1 0 0 0 0 1 1	_ 67	_ 67
$B_{ДК}=$	1.1 0 0 1 1 0 1	- 51	205
$C_{ПК}=$	0.1 1 1 0 1 1 0	118	118 ?

***CF* = 1; *SF* = 0; *ZF* = 0; *AF* = 1; *PF* = 0; *OF* = 0.**

Для БзИ результат неверен вследствие возникающего заема из разряда за пределами формата.

4) $A < 0$, $B < 0$

		ЗИ	БзИ
$A_{ДК}=$	_ 1.0 1 1 1 1 1 0 1	_ - 67	_ 189
$B_{ДК}=$	1.1 0 0 1 1 0 1	- 51	205
$C_{ДК}=$	1.1 1 1 0 0 0 0		240 ?
$C_{ПК}=$	1.0 0 1 0 0 0 0	-16	

***CF* = 1; *SF* = 1; *ZF* = 0; *AF* = 0; *PF* = 1; *OF* = 0.**

Для БзИ результат неверен вследствие возникающего заема из разряда за пределами формата.

3.2. Правило для подбора операнда ***B*** выглядит следующим образом:

$$A + B > 128, \text{ значит } 128 - A < B < 127$$

A = 67 (неизменно), ***B*** = 64 (подбираем)

1) $A < 0$, $B > 0$

		ЗИ	БзИ
$A_{ДК}=$	_ 1.0 1 1 1 1 1 0 1	_ - 67	_ 189
$B_{ПК}=$	0.1 0 0 0 0 0 0	64	64
$C_{ПК}=$	0.1 1 1 1 1 0 1	125?	125

***CF* = 0; *SF* = 0; *ZF* = 0; *AF* = 0; *PF* = 1; *OF* = 1.**

Для знакового вычитания результат некорректен вследствие переполнения, о котором можно судить по одному из двух признаков:

- при разных знаках операндов знак результата отличается от знака первого операнда;
- несовпадение заемов в два старшие разряда (один из них присутствует, а другой нет).

2) $A > 0$, $B < 0$

		ЗИ	БзИ
$A_{ПК}=$	_ 0.1 0 0 0 0 1 1	_ 67	_ 67
$B_{ДК}=$	1.1 0 0 0 0 0 0	- 64	192
$C_{ДК}=$	1.0 0 0 0 0 1 1		131 ?
$C_{ПК}=$	1.1 1 1 1 1 0 1	-125?	

***CF* = 1; *SF* = 1; *ZF* = 0; *AF* = 0; *PF* = 0; *OF* = 1.**

Для ЗИ результат некорректен вследствие возникающего переполнения, для БзИ результат некорректен из-за возникающего переноса из старшего разряда.

3.3. Значение числа ***B*** фиксируем (***B*** = 51), а значение ***A*** определяется из условия ***A* + *B*** = 128, благодаря которой при вычитании из положительного числа отрицательного будет фиксироваться переполнение, а при вычитании из отрицательного числа положительного не будет. ***A*** = 77.

1) ***A*** > 0, ***B*** < 0

		ЗИ	**БзИ**
$\boldsymbol{A}_{ПК}$ = _	0.1 0 0 1 1 0 1	_ 77	_ 77
$\boldsymbol{B}_{ДК}$ =	1.1 0 0 1 1 0 1	- 51	205
$\boldsymbol{C}_{ДК}$ =	1.0 0 0 0 0 0 0		128?
$\boldsymbol{C}_{ПК}$ =	1.0 0 0 0 0 0 0	- 128?	

***CF* = 1; *SF* = 1; *ZF* = 0; *AF* = 0; *PF* = 0; *OF* = 1.**

Результат БзИ некорректен вследствие возникающего заема из разряда за пределами формата; для ЗИ результат некорректен из-за переполнения.

2) ***A*** < 0, ***B*** > 0

		ЗИ	**БзИ**
$\boldsymbol{A}_{ДК}$ _	1.0 1 1 0 0 1 1	_ -77	_ 179
$\boldsymbol{B}_{ПК}$	0.0 1 1 0 0 1 1	51	51
$\boldsymbol{C}_{ДК}$	1.0 0 0 0 0 0 0		128
$\boldsymbol{C}_{ПК}$	1.0 0 0 0 0 0 0	- 128	

***CF* = 0; *SF* = 1; *ZF* = 0; *AF* = 0; *PF* = 0; *OF* = 0.**

Результаты ЗИ и БзИ корректны.

З А Д А Н И Е 4
УМНОЖЕНИЕ ЦЕЛЫХ ЧИСЕЛ

1. В разрядной сетке длиной в один байт (один разряд знаковый и семь – цифровых) выполнить операцию умножения заданных целых чисел ***A*** и ***B*** со всеми комбинациями знаков, используя метод умножения в дополнительных кодах с применением коррекции. При выполнении операции использовать способ умножения с поразрядным анализом множителя, начиная от его младших разрядов со сдвигом суммы частных произведений (СЧП) вправо. Для представления произведения использовать удвоенную разрядную сетку (16 двоичных разрядов: один – знаковый и 15 – цифровых). Результаты представить в десятичной системе счисления и проверить их правильность.

2. В той же разрядной сетке, что и в п.1, выполнить операцию умножения заданных целых чисел ***A*** и ***B*** со всеми комбинациями знаков, используя метод умножения в дополнительных кодах без коррекции. Результаты представить в десятичной системе счисления и проверить их правильность.

Варианты заданий приведены в табл.4 приложения 1.

О с н о в н ы е п о л о ж е н и я. Использование метода умножения в дополнительных кодах базируется на представлении отрицательных операндов и участии их в операции в дополнительном коде. В отличие от метода умножения в прямых кодах не требуется выполнять преобразование отрицательных операндов из дополнительного кода в прямой, а отрицательного результата – из прямого кода в дополнительный. Результат операции получается в коде, зависящем от знака, т.е. положительный – в прямом, а отрицательный – в дополнительном. Знаковые разряды операндов участвуют в операции умножения точно так же, как и цифровые. Это означает, что в сложении с СЧП вступают все разряды множимого, включая знаковый, и в анализе разрядов множителя с целью определения последующих действий над СЧП участвует знаковый разряд, т.е. на него производится умножение, как и на любой цифровой.

4.1. У м н о ж е н и е в д о п о л н и т е л ь н ы х к о д а х с п р и м е н е н и е м к о р р е к ц и и

При использовании традиционного метода умножения в дополнительных кодах только в случае положительных операндов результат получается в явном виде, в остальных же случаях он требует коррекции. Применяются два вида коррекции: а) коррекция в ходе перемножения операндов; б) коррекция окончательного результата. Коррекция первого вида имеет место при отрицательном множимом и состоит в модифицированном сдвиге СЧП вправо, при котором в освобождающийся старший разряд СЧП вносится единица. Коррекция второго вида производится при отрицательном множителе и состоит в вычитании множимого из старших разрядов СЧП, которое может сводиться к сложению с дополнением множимого.

При умножении на младшие нули множителя в случае отрицательного множимого сдвиг нулевой СЧП производится обычным образом (не модифицированный), т.е. в освобождающийся старший разряд вносится нуль.

Пример 1. $\boldsymbol{A} = 15$, $\boldsymbol{B} = 13$.

Для иллюстрации метода используется укороченная по сравнению с заданием разрядная сетка для операндов (один разряд знаковый и 4 – цифровых) и результата (один разряд знаковый и 9 – цифровых). При выполнении примеров выделен анализируемый на каждом шаге разряд множителя, а также показано последовательное вытеснение множителя при его сдвиге вправо и заполнение его освобождающихся старших разрядов младшими разрядами СЧП. Таким образом, в начале операции СЧП занимает пять двоичных разрядов, а в конце – результат представлен десятью разрядами.

Представление операндов в разрядной сетке:

$[+\boldsymbol{A}]_{\boldsymbol{пр}} = 0.1111$; $[-\boldsymbol{A}]_{\boldsymbol{доп}} = 1.0001$;

$[+\boldsymbol{B}]_{\boldsymbol{пр}} = 0.1101$; $[-\boldsymbol{B}]_{\boldsymbol{доп}} = 1.0011$.

а) Множимое отрицательное ($\boldsymbol{A} < 0$), множитель положительный ($\boldsymbol{B} > 0$):

№ шага	Операнды и действия	СЧП (старшие разряды)	Множитель и СЧП (младшие разряды)	Пояснения
1	2	3	4	5
0	*СЧП*	0 0 0 0 0	$0\,1\,1\,0\,\boxed{1}$	Обнуление старших разрядов СЧП
1	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{1\,0\,0\,0\,1}$ 1 0 0 0 1 1 1 0 0 0	 0 1 1 0 1 $1 \mid 0\,1\,1\,\boxed{0}$	Сложение СЧП с множимым Модифицированный сдвиг СЧП и множителя вправо
2	$\overrightarrow{СЧП}$	1 1 1 0 0	$0\,1 \mid 0\,1\,\boxed{1}$	Модифицированный сдвиг СЧП и множителя вправо
3	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{1\,0\,0\,0\,1}$ 0 1 1 0 1 1 0 1 1 0	 $0\,1 \mid 0\,1\,1$ $1\,0\,1 \mid 0\,\boxed{1}$	Сложение СЧП с множимым Модифицированный сдвиг СЧП и множителя вправо
4	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{1\,0\,0\,0\,1}$ 0 0 1 1 1 1 0 0 1 1	 $1\,0\,1 \mid 0\,1$ $1\,1\,0\,1 \mid \boxed{0}$	Сложение СЧП с множимым Модифицированный сдвиг СЧП и множителя вправо
5	$\overrightarrow{СЧП}$	1 1 0 0 1	1 1 1 0 1	Модифицированный сдвиг СЧП и множителя вправо

Полученный результат отрицателен и представлен в дополнительном коде:

$[\boldsymbol{C}]_{доп} = [\boldsymbol{A}]_{доп} \times [\boldsymbol{B}]_{пр} = (1.100111101)_2.$

Для проверки правильности результата переведем его в прямой код:

$[\boldsymbol{C}]_{пр} = (1.011000011)_2 = (–195)_{10}.$

б) $\boldsymbol{A} > 0$, $\boldsymbol{B} < 0$:

1	2	3	4	5
0	*СЧП*	0 0 0 0 0	1 0 0 1 $\boxed{1}$	Обнуление старших разрядов СЧП
1	$[A]_{пр}$ *СЧП* $\overrightarrow{СЧП}$	<u>0 1 1 1 1</u> 0 1 1 1 1 0 0 1 1 1	 1 0 0 1 1 1 | 1 0 0 $\boxed{1}$	Сложение СЧП с множимым Сдвиг СЧП и множителя вправо
2	$[A]_{пр}$ *СЧП* $\overrightarrow{СЧП}$	<u>0 1 1 1 1</u> 1 0 1 1 0 0 1 0 1 1	 1 | 1 0 0 1 0 1 | 1 0 $\boxed{0}$	Сложение СЧП с множимым Сдвиг СЧП и множителя вправо
3	$\overrightarrow{СЧП}$	0 0 1 0 1	1 0 1 | 1 $\boxed{0}$	Сдвиг СЧП и множителя вправо
4	$\overrightarrow{СЧП}$	0 0 0 1 0	1 1 0 1 | $\boxed{1}$	Сдвиг СЧП и множителя вправо
5	$[A]_{пр}$ *СЧП* $\overrightarrow{СЧП}$	<u>0 1 1 1 1</u> 1 0 0 0 1 0 1 0 0 0	 1 1 0 1 | 1 1 1 1 0 1	Сложение СЧП с множимым Сдвиг СЧП и множителя вправо
6	$[–A]_{доп}$ *СЧП*	<u>1 0 0 0 1</u> 1 1 0 0 1	 1 1 1 0 1	Коррекция результата: сложение старших разрядов СЧП с дополнением множимого

Полученный результат отрицателен и представлен в дополнительном коде:

$[\boldsymbol{C}]_{доп} = [\boldsymbol{A}]_{пр} \times [\boldsymbol{B}]_{доп} = (1.100111101)_2$,

$[\boldsymbol{C}]_{пр} = (1.011000011)_2 = (–195)_{10}.$

в) Оба операнда отрицательные ($\boldsymbol{A} < 0$, $\boldsymbol{B} < 0$):

1	2	3	4	5
0	*СЧП*	0 0 0 0 0	1 0 0 1 $\boxed{1}$	Обнуление старших разрядов СЧП
1	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	<u>1 0 0 0 1</u> 1 0 0 0 1 1 1 0 0 0	 1 0 0 1 1 1 | 1 0 0 $\boxed{1}$	Сложение СЧП с множимым Модифицированный сдвиг СЧП и множителя вправо
2	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	<u>1 0 0 0 1</u> 0 1 0 0 1 1 0 1 0 0	 1 | 1 0 0 1 1 1 | 1 0 $\boxed{0}$	Сложение СЧП с множимым Модифицированный сдвиг СЧП и множителя вправо
3	$\overrightarrow{СЧП}$	1 1 0 1 0	0 1 1 | 1 $\boxed{0}$	Модифицированный сдвиг СЧП и множителя вправо
4	$\overrightarrow{СЧП}$	1 1 1 0 1	0 0 1 1 | $\boxed{1}$	Модифицированный сдвиг СЧП и множителя вправо
5	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	<u>1 0 0 0 1</u> 0 1 1 1 0 1 0 1 1 1	 0 0 1 1 | 1 0 0 0 1 1	Сложение СЧП с множимым Модифицированный сдвиг СЧП и множителя вправо
6	$[–\boldsymbol{A}]_{пр}$ *СЧП*	<u>0 1 1 1 1</u> 0 0 1 1 0	 0 0 0 1 1	Коррекция результата: сложение старших разрядов СЧП с дополнением множимого

Полученный результат положителен и представлен в прямом коде:
$[\boldsymbol{C}]_{пр} = [\boldsymbol{A}]_{доп} \times [\boldsymbol{B}]_{доп} = (0.011000011)_2 = (195)_{10}$.

4.2. Умножение в дополнительных кодах без коррекции

Наряду с традиционным методом умножения в дополнительных кодах, требующим коррекции результата, достаточно широкое применение в ЭВМ находит метод Бута, при котором не требуется выполнять коррекцию. Особенность метода состоит в выполнении сложения или вычитания СЧП и множимого на каждом шаге умножения в зависимости от того, как после сдвига вправо изменяется младший разряд множителя. При его изменении с единицы на ноль производится сложение СЧП с множимым, а при изменении с нуля на единицу – вычитание множимого из СЧП, которое может быть реализовано как сложение с дополнением множимого. Если младший разряд множителя при сдвиге не изменяется, то на данном шаге не производится сложения (вычитания), а выполняется только сдвиг СЧП и множителя вправо.

При реализации этого метода происходит чередование сложений и вычитаний множимого и СЧП, вследствие чего старший разряд СЧП в явном виде представляет его знак. При сдвиге СЧП вправо значение знакового разряда сохраняется (арифметический сдвиг).

Необходимо отметить, что:
а) При умножении на младшую единицу множителя производится вычитание множимого из СЧП, поскольку считается, что происходит изменение младшего разряда множителя с нуля на единицу.
б) При умножении на младшие нули множителя осуществляется сдвиг нулевой СЧП и множителя вправо до появления единицы в младшем разряде множителя, после чего производится вычитание множимого из СЧП.

Пример 2. $\boldsymbol{A} = 11, \boldsymbol{B} = 15$.

Представление операндов в разрядной сетке:

$[+\boldsymbol{A}]_{пр} = 0.1011$; $[-\boldsymbol{A}]_{доп} = 1.0101$;
$[+\boldsymbol{B}]_{пр} = 0.1111$; $[-\boldsymbol{B}]_{доп} = 1.0001$.

а) Оба операнда положительные (***A*** > 0, ***B*** > 0):

1	2	3	4	5
0	*СЧП*	0 0 0 0 0	0 1 1 1 $\boxed{1}$	Обнуление старших разрядов СЧП
1	$[-A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{1\ 0\ 1\ 0\ 1}$ 1 0 1 0 1 1 1 0 1 0	 0 1 1 1 1 1 $\mid$ 0 1 1 $\boxed{1}$	Младший разряд множителя равен 1: вычитание множимого из СЧП Сдвиг СЧП и множителя вправо
2	$\overrightarrow{СЧП}$	1 1 1 0 1	0 1 $\mid$ 0 1 $\boxed{1}$	При сдвиге младший разряд не изменился Сдвиг СЧП и множителя вправо
3	$\overrightarrow{СЧП}$	1 1 1 1 0	1 0 1 $\mid$ 0 $\boxed{1}$	При сдвиге младший разряд не изменился Сдвиг СЧП и множителя вправо
4	$\overrightarrow{СЧП}$	1 1 1 1 1	0 1 0 1 $\mid$ $\boxed{0}$	При сдвиге младший разряд не изменился Сдвиг СЧП и множителя вправо
5	$[A]_{пр}$ СЧП $\overrightarrow{СЧП}$	$\underline{0\ 1\ 0\ 1\ 1}$ 0 1 0 1 0 0 0 1 0 1	 0 1 0 1 $\mid$ 0 0 0 1 0 1	При сдвиге младший разряд множителя изменился с 1 на 0: сложение СЧП с множимым Сдвиг СЧП и множителя вправо

Полученный результат представлен в прямом коде и равен:

$[\boldsymbol{C}]_{\boldsymbol{пр}} = 2^0 + 2^2 + 2^5 + 2^7 = 1 + 4 + 32 + 128 = 165.$

б) Оба операнда отрицательные (***A*** < 0, ***B*** < 0):

1	2	3	4	5
0	*СЧП*	0 0 0 0 0	1 0 0 0 $\boxed{1}$	Обнуление старших разрядов СЧП
1	$[-A]_{пр}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{0\ 1\ 0\ 1\ 1}$ 0 1 0 1 1 0 0 1 0 1	 1 0 0 0 1 1 $\mid$ 1 0 0 $\boxed{0}$	Вычитание множимого из СЧП Сдвиг СЧП и множителя вправо
2 П	$[A]_{доп}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{1\ 0\ 1\ 0\ 1}$ 1 1 0 1 0 1 1 1 0 1	 1 $\mid$ 1 0 0 0 0 1 $\mid$ 1 0 $\boxed{0}$	Сложение СЧП с множимым Сдвиг СЧП и множителя вправо
3	$\overrightarrow{СЧП}$	1 1 1 1 0	1 0 1 $\mid$ 1 $\boxed{0}$	Сдвиг СЧП и множителя вправо
4	$\overrightarrow{СЧП}$	1 1 1 1 1	0 1 0 1 $\mid$ $\boxed{1}$	Сдвиг СЧП и множителя вправо
5	$[-A]_{пр}$ *СЧП* $\overrightarrow{СЧП}$	$\underline{0\ 1\ 0\ 1\ 1}$ 0 1 0 1 0 0 0 1 0 1	 0 1 0 1 $\mid$ 1 0 0 1 0 1	Сложение СЧП с множимым Сдвиг СЧП и множителя вправо

Полученный результат положителен и представлен в прямом коде:

$[\boldsymbol{C}]_{\boldsymbol{пр}} = [\boldsymbol{A}]_{\boldsymbol{доп}} \times [\boldsymbol{B}]_{\boldsymbol{доп}} = (0.010100101)_2 = (165)_{10}$

ЗАДАНИЕ 5
ДЕЛЕНИЕ ЦЕЛЫХ ЧИСЕЛ

1. Выполнить операцию деления заданных целых чисел ***A*** и ***B*** со всеми комбинациями знаков, используя метод деления в дополнительных кодах. Для представления делимого (***A***) использовать 16 двоичных разрядов (один –знаковый и 15 – цифровых), для представления делителя (***B***) – 8 разрядов (один – знаковый и 7 – цифровых). Остаток от деления и частное представляются в той же разрядной сетке, что и делитель.
2. Результаты операции представить в десятичной системе счисления и проверить их правильность.

Варианты задания приведены в табл.5 приложения 1.

Основные положения. Основная особенность операции деления целых чисел в ЭВМ состоит в том, что делимое (первый операнд) представляется в удвоенной разрядной сетке по сравнению с делителем (вторым операндом). В качестве результата операции формируются частное и остаток, представляемые в той же разрядной сетке, что и делитель, причем остаток замещает старшие разряды делимого, а частное – младшие.

Использование метода деления в дополнительных кодах предполагает, что положительные операнды участвуют в операции в прямом, а отрицательные – в дополнительном коде. Результат операции (остаток и частное) получается в коде, зависящем от знака. Знак частного определяется знаками делимого и делителя и формируется в ходе выполнения операции по тем же правилам, что и цифровые разряды. Знак остатка должен совпадать со знаком делимого. Нулевые частное и остаток содержат в знаковом разряде ноль («положительный» ноль) независимо от знаков операндов.

Формирование частного. Операция деления при ее реализации в ЭВМ представляет собой многошаговый процесс, на каждом шаге которого формируется один разряд частного (знаковый или цифровой). Значение разряда частного зависит от знаков делителя и остатка, полученного на данном шаге. При их совпадении формируемый разряд частного равен единице, при несовпадении – нулю.

Разряды частного заносятся на место освобождающихся при сдвиге влево младших разрядов делимого (остатка). Частное формируется, начиная от старших разрядов. На первом шаге деления вырабатывается его знаковый разряд, а на последующих – цифровые разряды.

Действия над остатком. На каждом шаге деления производится удвоение остатка, полученного на предыдущем шаге, посредством его сдвига на один разряд влево (на первом шаге производится сдвиг делимого) и формирование нового остатка путем сложения старших разрядов с делителем или вычитания делителя из старших разрядов остатка. Вычитание делителя, как правило, заменяется сложением с его дополнением. Действие, выполняемое над остатком, определяется знаками остатка и делителя. При их совпадении производится вычитание делителя из остатка, при несовпадении – делитель прибавляется к остатку.

Следует помнить, что:

а) При сдвиге остатка (делимого) влево в освобождающийся младший разряд заносится ноль, а затем (после выполнения соответствующего действия над остатком) на этом месте формируется разряд частного.

б) В качестве знака остатка, по которому определяется последующее действие над ним (сложение с делителем или вычитание делителя), используется значение знакового разряда остатка полученного на предыдущем шаге, т.е. до его сдвига влево. При сдвиге остатка влево это значение может измениться в случае несовпадения знакового и старшего цифрового разрядов остатка до сдвига. Использование модифицированного кода (с двумя знаковыми разрядами) для представления остатка позволяет упростить процедуру анализа его знака, так как при сдвиге влево может быть искажен только младший (правый) знаковый разряд, старший же (левый) знаковый разряд будет представлять истинное значение знака остатка как до, так и после сдвига, и по нему можно определить последующее действие над остатком.

Коррекция результата. Результат операции деления (остаток и частное) может потребовать коррекции. Коррекция остатка производится в случае несовпадения знака окончательного остатка, полученного после формирования всех разрядов частного, со знаком делимого и состоит в сложении остатка с делителем или в вычитании делителя из него. Действие, выполняемое при коррекции остатка, определяется так же, как и в основном цикле деления, связанном с формированием разрядов частного, т.е. зависит от совпадения или несовпадения знаков делителя и остатка. Исключением из этого правила является нулевой остаток, который независимо от знака делимого не подлежит коррекции.

Кроме общего правила коррекции остатка, существует особый случай, который имеет место при получении нулевого промежуточного остатка на каком-либо шаге деления. На следующем шаге к нулевому остатку, который имеет знак “+”, будет прибавлен отрицательный делитель или его дополнение при положительном делителе. При сдвиге полученного остатка влево на последующем шаге его величина удвоится. Сложение отрицательного остатка с положительным делителем (или с дополнением отрицательного делителя) снова дает отрицательный остаток, по модулю равный величине делителя. Если же в этом случае делимое отрицательное, то получаемый в конце операции отрицательный остаток не может быть скорректирован по общему правилу, так как имеет знак, совпадающий со знаком делимого. Вследствие этого общее правило коррекции остатка дополняется следующим: при отрицательных остатке и делимом производится сравнение модулей остатка и делителя и, если они совпадают, осуществляется коррекция остатка по общему правилу, после чего остаток становится нулевым.

Практическая реализация этого случая на ЭВМ может быть осуществлена следующим образом. При получении отрицательного окончательного остатка в случае отрицательного делимого делается попытка скорректировать полученный остаток. Если в результате коррекции получается нулевой остаток, он является правильным. При получении ненулевого остатка он восстанавливается путем сложения с делителем (если делитель положительный) или с его дополнением (если делитель отрицательный).

Коррекция частного производится при получении нулевого остатка в случае отрицательного делимого и состоит в вычитании единицы из отрицательного частного (делитель положительный) или в прибавлении единицы к положительному частному (делитель отрицательный)

Проверка корректности деления. Одной из особенностей операции деления целых чисел, связанной с представлением делимого в удвоенном по сравнению с делителем формате, является возможность получения такого частного, которое не помещается в отведенный для него формат. Этот случай классифицируется как «некорректность деления с фиксированной запятой» и является причиной прерывания выполняемой программы. В связи с этим при делении целых чисел производится проверка корректности результата, осуществляемая в начале операции.

При использовании метода деления и дополнительных кодах условие корректности деления имеет вид:

а) $|\boldsymbol{A}| / |\boldsymbol{B}| < 2^n$, при одинаковых знаках операндов;

б) $|\boldsymbol{A}| / |\boldsymbol{B}| < 2^n + 1$, при разных знаках операндов,

где n – число цифровых разрядов частного (делителя).

Правая часть этих неравенств определяется максимальным по модулю целым числом, которое можно представить в формате с использованием n цифровых разрядов (с учетом знакового разряда формат частного составляет n+1 разрядов). При этом предполагается, что положительное частное представляется в прямом коде (случай а), а отрицательное – в дополнительном (случай б).

Для проверки корректности деления вычисляется разность:

а) $|\boldsymbol{A}| - |\boldsymbol{B}| \times 2^n$, при одинаковых знаках операндов;

б) $|\boldsymbol{A}| - |\boldsymbol{B}| - |\boldsymbol{B}| \times 2^n$, при разных знаках операндов – и в зависимости от знака этой разности устанавливается корректность или некорректность деления.

Вследствие того, что при делении в дополнительных кодах операнды участвуют в операции вместе со своими знаками, вычитание *производится только в случае одинаковых знаков операндов. При разных знаках операндов вычитание заменяется сложением.*

Фактически проверка корректности деления, выполняемая путем так называемого «пробного вычитания», совмещается с первым шагом деления, на котором вырабатывается знаковый разряд частного. На этом шаге при одинаковых знаках операндов производится сдвиг делимого на один разряд влево, и затем вычитание делителя (сложение его с дополнением) из его старших разрядов. Знак остатка, полученного на первом шаге, сравнивается со знаком делимого, и если они совпадают, то процесс деления завершается ввиду некорректности результата.

При разных знаках операндов на первом шаге производится сложение делимого и делителя, выровненных по младшим разрядам, затем сдвиг полученного остатка влево, после чего осуществляется его сложение с делителем, выровненным по старшим разрядам. Знак полученного после этого сложения остатка сравнивается со знаком делимого, и если они совпадают, деление является некорректным.

Особенности проверки корректности деления:

а) Нулевой остаток содержит в знаковом разряде ноль и поэтому рассматривается как положительный.

б) При сложении делимого с делителем, выровненным по младшим разрядам, старшие разряды делителя дополняются его знаковым разрядом (расширение знака на старшие разряды).

в) Некорректность деления сказывается на неправильном формировании знака частного на первом шаге деления. Действительно, при одинаковых знаках операндов частное должно иметь знак “+”, однако при совпадении знака первого остатка со знаком делимого (и следовательно делителя) в качестве знакового разряда частного сформируется единица (знак “–”). Аналогично при разных знаках операндов частное должно иметь знак “–”, но при совпадении знаков остатка и делимого (и, следовательно, несовпадении знаков остатка и делителя) в качестве знакового разряда сформируется ноль (знак “+”).

Пример 1. Деление с ненулевым остатком.

$\boldsymbol{A} = 139, \quad \boldsymbol{B} = 13.$

Для иллюстрации метода используется укороченная по сравнению с заданием разрядная сетка: для делимого 10 разрядов (один знаковый и 9 – цифровых), для делителя, остатка и частного – 5 разрядов (один знаковый и 4 – цифровых).

Представление операндов с разными знаками в разрядной сетке:

$[+\boldsymbol{A}]_{пр} = 0.010001011; \quad [-\boldsymbol{A}]_{доп} = 1.101110101;$

$[+\boldsymbol{B}]_{пр} = 0.1101; \quad [-\boldsymbol{B}]_{доп} = 1.0011.$

а) Делимое положительное ($\boldsymbol{A} > 0$), делитель отрицательный ($\boldsymbol{B} < 0$):

<table>
<tr><th>№ шага</th><th>Операнды и дейст-вия</th><th>Делимое и остаток (старшие разряды)</th><th>Делимое и остаток (младшие разряды), частное</th><th>Пояснения</th></tr>
<tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr>
<tr><td>0</td><td>$[A]_{пр}$</td><td>0 0 1 0 0</td><td>0 1 0 1 1</td><td>Делимое</td></tr>
<tr><td>1</td><td>$[B]_{доп}$
R'_1
$\overleftarrow{R}'_1$
$[B]_{доп}$
R_1</td><td>1 1 1 1 1
0 0 0 1 1
0 0 1 1 1
+
1 0 0 1 1
$\boxed{1}$ 1 0 1 0
1 1 0 1 0
ЗнR_1=ЗнB</td><td>1 0 0 1 1
1 1 1 1 0
1 1 1 0 | 0
1 1 1 0 | $\boxed{1}$</td><td>Сложение с делителем, выровненным по младшим разрядам
Сдвиг остатка влево
Сложение с делителем, выровненным по старшим разрядам
Знак первого остатка не совпадает со знаком делимого–деление корректно
Формирование знака частного</td></tr>
<tr><td>2</td><td>$\overleftarrow{R}_1$
$[-B]_{пр}$
R_2</td><td>1 0 1 0 1
+
0 1 1 0 1
0 0 0 1 0
ЗнR_2≠ЗнB</td><td>1 1 0 | 1 0
1 1 0 | 1 $\boxed{0}$</td><td>Сдвиг остатка влево
Вычитание делителя
Формирование цифры частного</td></tr>
<tr><td>3</td><td>$\overleftarrow{R}_2$
$[B]_{доп}$
R_3</td><td>+ 0 0 1 0 1
1 0 0 1 1
1 1 0 0 0
ЗнR_3=ЗнB</td><td>1 0 | 1 0 0
1 0 | 1 0 $\boxed{1}$</td><td>Сдвиг остатка влево
Сложение с делителем
Формирование цифры частного</td></tr>
<tr><td>4</td><td>$\overleftarrow{R}_3$
$[-B]_{пр}$
R_4</td><td>1 0 0 0 1
+
0 1 1 0 1
1 1 1 1 0
ЗнR_4=ЗнB</td><td>0 | 1 0 1 0
0 | 1 0 1 $\boxed{1}$</td><td>Сдвиг остатка влево
Вычитание делителя
Формирование цифры частного</td></tr>
<tr><td>5</td><td>$\overleftarrow{R}_4$
$[-B]_{пр}$
R_5</td><td>1 1 1 0 0
+
0 1 1 0 1
0 1 0 0 1
ЗнR_5≠ЗнB</td><td>1 0 1 1 0
1 0 1 1 $\boxed{0}$</td><td>Сдвиг остатка влево
Вычитание делителя
Формирование цифры частного</td></tr>
</table>

В результате выполнения операции получено отрицательное частное $[\boldsymbol{C}]_{доп} = (1.0110)_2$, $[\boldsymbol{C}]_{пр} = (1.1010)_2 = (-10)_{10}$ и положительный остаток $[\boldsymbol{R}]_{пр} = (0.1001)_2 = (+9)_{10}$, которые соответствуют истинным значениям: $(-10) \times (-13) + 9 = 139$.

б) Делимое отрицательное ($\boldsymbol{A} < 0$), делитель положительный ($\boldsymbol{B} > 0$):

1	2	3	4	5
0	$[A]_{доп}$	1 1 0 1 1	1 0 1 0 1	Делимое
1	$[B]_{пр}$ R_1' $\overleftarrow{R}_1'$ $[B]_{пр}$ R_1	0 0 0 0 0 1 1 1 0 0 1 1 0 0 0 + 0 1 1 0 1 $\boxed{0}$ 0 1 0 1 0 0 1 0 1 ЗнR_1=ЗнB	0 1 1 0 1 0 0 0 1 0 0 0 1 0 | 0 0 0 1 0 | $\boxed{1}$	Сложение с делителем, выровненным по младшим разрядам Сдвиг остатка влево Сложение с делителем, выровненным по старшим разрядам Знак первого остатка не совпадает со знаком делимого–деление корректно Формирование знака частного
2	$\overleftarrow{R}_1$ $[-B]_{доп}$ R_2	+ 0 1 0 1 0 1 0 0 1 1 1 1 1 0 1 ЗнR_2≠ЗнB	0 1 0 | 1 0 0 1 0 | 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
3	$\overleftarrow{R}_2$ $[B]_{пр}$ R_3	+ 1 1 0 1 0 0 1 1 0 1 0 0 1 1 1 ЗнR_3=ЗнB	1 0 | 1 0 0 1 0 | 1 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
4	$\overleftarrow{R}_3$ $[-B]_{доп}$ R_4	+ 0 1 1 1 1 1 0 0 1 1 0 0 0 1 0 ЗнR_4=ЗнB	0 | 1 0 1 0 0 | 1 0 1 $\boxed{1}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
5	$\overleftarrow{R}_4$ $[-B]_{доп}$ R_5	+ 0 0 1 0 0 1 0 0 1 1 1 0 1 1 1 ЗнR_5≠ЗнB	1 0 1 1 0 1 0 1 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного

В результате выполнения операции получено отрицательное частное $[\boldsymbol{C}]_{доп} = (1.0110)_2$, $[\boldsymbol{C}]_{пр} = (1.1010)_2 = (-10)_{10}$ и отрицательный остаток $[\boldsymbol{R}]_{доп} = (1.0111)_2$, $[\boldsymbol{R}]_{пр} = (1.1001)_2 = (-9)_{10}$, которые соответствуют истинным значениям: $(-10) \times 13 + (-9) = -139$.

в) Оба операнда отрицательные ($\boldsymbol{A} < 0$, $\boldsymbol{B} < 0$):

1	2	3	4	5
0	$[A]_{доп}$	1 1 0 1 1	1 0 1 0 1	Делимое
1	$[\overleftarrow{A}]_{доп}$ $[-B]_{пр}$ R_1	+ 1 0 1 1 1 0 1 1 0 1 $\boxed{0}$ 0 1 0 0 0 0 1 0 0 ЗнR_1≠ЗнB	0 1 0 1 | 0 0 1 0 1 | $\boxed{0}$	Сдвиг делимого влево Вычитание делителя Знак первого остатка не совпадает со знаком делимого–деление корректно Формирование знака частного
2	$\overleftarrow{R}_1$ $[B]_{доп}$ R_2	+ 0 1 0 0 0 1 0 0 1 1 1 1 0 1 1 ЗнR_2=ЗнB	1 0 1 | 0 0 1 0 1 | 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
3	$\overleftarrow{R}_2$ $[-B]_{пр}$ R_3	+ 1 0 1 1 1 0 1 1 0 1 0 0 1 0 0 ЗнR_3≠ЗнB	0 1 | 0 1 0 0 1 | 0 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
4	$\overleftarrow{R}_3$ $[B]_{доп}$ R_4	+ 0 1 0 0 0 1 0 0 1 1 1 1 0 1 1 ЗнR_4=ЗнB	1 | 0 1 0 0 1 | 0 1 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
5	$\overleftarrow{R}_4$ $[-B]_{пр}$ R_5	+ 1 0 1 1 1 0 1 1 0 1 0 0 1 0 0 ЗнR_5≠ЗнB	0 1 0 1 0 0 1 0 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
6	$[B]_{доп}$ R_6	1 0 0 1 1 1 0 1 1 1	 0 1 0 1 0	Коррекция остатка: сложение с делителем Результат

В результате выполнения операции получено положительное частное $[\boldsymbol{C}]_{пр} = (0.1010)_2 = (+10)_{10}$ и отрицательный остаток $[\boldsymbol{R}]_{доп} = (1.0111)_2$, $[\boldsymbol{R}]_{пр} = (1.1001)_2 = (-9)_{10}$, которые соответствуют истинным значениям: $10 \times (-13) - 9 = -139$.

Пример 2. Деление с нулевым остатком.

$\boldsymbol{A} = 72$, $\boldsymbol{B} = 6$.

Представление операндов в разрядной сетке.

$[+\boldsymbol{A}]_{пр} = 0.001001000$; $[-\boldsymbol{A}]_{доп} = 1.110111000$;

$[+\boldsymbol{B}]_{пр} = 0.0110$; $[-\boldsymbol{B}]_{доп} = 1.1010$.

а) Делимое отрицательное ($\boldsymbol{A} < 0$), делитель положительный ($\boldsymbol{B} > 0$):

1	2	3	4	5
0	$[A]_{доп}$	1 1 1 0 1	1 1 0 0 0	Делимое
1	$[B]_{пр}$ R_1' $\overleftarrow{R}_1'$ $[B]_{пр}$ R_1	0 0 0 0 0 1 1 1 0 1 1 1 0 1 1 + 0 0 1 1 0 $\boxed{0}$ 0 0 0 1 0 0 0 0 1 ЗнR_1=ЗнB	0 0 1 1 0 1 1 1 1 0 1 1 1 0 | 0 1 1 1 0 | $\boxed{1}$	Сложение с делителем, выровненным по младшим разрядам Сдвиг остатка влево Сложение с делителем, выровненным по старшим разрядам Знак первого остатка не совпадает со знаком делимого–деление корректно Формирование знака частного
2	$\overleftarrow{R}_1$ $[-B]_{доп}$ R_2	+ 0 0 0 1 1 1 1 0 1 0 1 1 1 0 1 ЗнR_2≠ЗнB	1 1 0 | 1 0 1 1 0 | 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
3	$\overleftarrow{R}_2$ $[B]_{пр}$ R_3	+ 1 1 0 1 1 0 0 1 1 0 0 0 0 0 1 ЗнR_3=ЗнB	1 0 | 1 0 0 1 0 | 1 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
4	$\overleftarrow{R}_3$ $[-B]_{доп}$ R_4	+ 0 0 0 1 1 1 1 0 1 0 1 1 1 0 1 ЗнR_4≠ЗнB	0 | 1 0 1 0 0 | 1 0 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
5	$\overleftarrow{R}_4$ $[B]_{пр}$ R_5	+ 1 1 0 1 0 0 0 1 1 0 0 0 0 0 0 ЗнR_5=ЗнB	1 0 1 0 0 1 0 1 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
6	$[-1]_{доп}$	 0 0 0 0 0	1 1 1 1 1 1 0 1 0 0	Коррекция частного: вычитание единицы Результат

В результате выполнения операции получено отрицательное частное

$[\boldsymbol{C}]_{доп} = (1.0100)_2$, $[\boldsymbol{C}]_{пр} = (1.2100) = (-12)_{10}$ и нулевой остаток.

б) Оба операнда отрицательные ($\boldsymbol{A} < 0$, $\boldsymbol{B} < 0$):

1	2	3	4	5
0	$[A]_{доп}$	1 1 1 0 1	1 1 0 0 0	Делимое
1	$[\overleftarrow{A}]_{доп}$ $[-B]_{пр}$ R_1	+ 1 1 0 1 1 0 0 1 1 0 $\boxed{0}$ 0 0 0 1 0 0 0 0 1 ЗнR_1≠ЗнB	1 0 0 0 | 0 1 0 0 0 | $\boxed{0}$	Сдвиг делимого влево Вычитание делителя Знак первого остатка не совпадает со знаком делимого–деление корректно Формирование знака частного
2	$\overleftarrow{R}_1$ $[B]_{доп}$ R_2	+ 0 0 0 1 1 1 1 0 1 0 1 1 1 0 1 ЗнR_2≠ЗнB	0 0 0 | 0 0 0 0 0 | 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
3	$\overleftarrow{R}_2$ $[-B]_{пр}$ R_3	+ 1 1 0 1 0 0 0 1 1 0 0 0 0 0 0 ЗнR_3≠ЗнB	0 0 | 0 1 0 0 0 | 0 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
4	$\overleftarrow{R}_3$ $[B]_{доп}$ R_4	+ 0 0 0 0 0 1 1 0 1 0 1 1 0 1 0 ЗнR_4≠ЗнB	0 | 0 1 0 0 0 | 0 1 0 $\boxed{1}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
5	$\overleftarrow{R}_4$ $[-B]_{пр}$ R_5	+ 1 0 1 0 0 0 0 1 1 0 1 1 0 1 0 ЗнR_5≠ЗнB	0 1 0 1 0 0 1 0 1 $\boxed{1}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
6	$[-B]_{пр}$ R_6	0 0 1 1 0 0 0 0 0 0	 0 1 0 1 1	Коррекция остатка, совпадающего с делителем: вычитание делителя
7	$[+1]_{пр}$	 0 0 0 0 0	0 0 0 0 1 0 1 1 0 0	Коррекция частного: сложение с единицей Результат

В результате выполнения операции получено положительное частное

$[\boldsymbol{C}]_{\boldsymbol{пр}} = (0.1100)_2 = (+12)_{10}$ и нулевой остаток.

Пример 3. Получение максимального по модулю частного и фиксация некорректности деления.

$\boldsymbol{A} = 254, \quad \boldsymbol{B} = 15.$

Представление операндов в разрядной сетке.

$[+\boldsymbol{A}]_{пр} = 0.011111110;$ $\qquad [-\boldsymbol{A}]_{доп} = 1.100000010;$

$[+\boldsymbol{B}]_{пр} = 0.1111;$ $\qquad [-\boldsymbol{B}]_{доп} = 1.0001.$

а) Оба операнда положительные ($\boldsymbol{A} > 0$, $\boldsymbol{B} > 0$):

1	2	3	4	5
0	$[A]_{пр}$	0 0 1 1 1	1 1 1 1 0	Делимое
1	$[\overleftarrow{A}]_{пр}$ $[-B]_{доп}$ R_1	+ 0 1 1 1 1 1 0 0 0 1 $\boxed{0}$ 0 0 0 0	1 1 1 0 $\vert$ 0	Сдвиг делимого влево Вычитание делителя Знак первого остатка совпадает со знаком делимого – деление некорректно

б) Делимое положительное ($\boldsymbol{A} > 0$), делитель отрицательный ($\boldsymbol{B} < 0$):

1	2	3	4	5
0	$[A]_{пр}$	0 0 1 1 1	1 1 1 1 0	Делимое
1	$[B]_{доп}$ R_1' $\overleftarrow{R}_1'$ $[B]_{доп}$ R_1	1 1 1 1 1 0 0 1 1 1 + 0 1 1 1 0 1 0 0 0 1 $\boxed{1}$ 1 1 1 1 1 1 1 1 1 ЗнR_1=ЗнB	1 0 0 0 1 0 1 1 1 1 1 1 1 1 $\vert$ 0 1 1 1 1 $\vert$ $\boxed{1}$	Сложение с делителем, выровненным по младшим разрядам Сдвиг остатка влево Сложение с делителем, выровненным по старшим разрядам Знак первого остатка не совпадает со знаком делимого – деление корректно Формирование знака частного
2	$\overleftarrow{R}_1$ $[-B]_{пр}$ R_2	+ 1 1 1 1 1 0 1 1 1 1 0 1 1 1 0 ЗнR_2≠ЗнB	1 1 1 $\vert$ 1 0 1 1 1 $\vert$ 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
3	$\overleftarrow{R}_2$ $[B]_{доп}$ R_3	+ 1 1 1 0 1 1 0 0 0 1 0 1 1 1 0 ЗнR_3≠ЗнB	1 1 $\vert$ 1 0 0 1 1 $\vert$ 1 0 $\boxed{0}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
4	$\overleftarrow{R}_3$ $[B]_{доп}$ R_4	+ 1 1 1 0 1 1 0 0 0 1 0 1 1 1 0 ЗнR_4≠ЗнB	1 $\vert$ 1 0 0 0 1 $\vert$ 1 0 0 $\boxed{0}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
5	$\overleftarrow{R}_4$ $[B]_{доп}$ R_5	+ 1 1 1 0 1 1 0 0 0 1 0 1 1 1 0 ЗнR_5≠ЗнB	1 0 0 0 0 1 0 0 0 $\boxed{0}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного

В результате выполнения операции получено отрицательное частное $[\boldsymbol{C}]_{доп} = (1.0000)_2 = (-16)_{10}$ и положительный остаток $[\boldsymbol{R}]_{пр} = (0.1110)_2 = (+14)_{10}$, которые соответствуют истинным значениям:

$(-16) \times (-15) + 14 = 254.$

В данном примере на 3-ем, 4-ом и 5-ом шагах выполняется сложение с отрицательным делителем, так как остаток, полученный на предыдущем шаге, положителен, но в результате сдвига влево его знак оказался искаженным. Для того, чтобы знак остатка при сдвиге влево не искажался, может быть использован модифицированный код (см. следующий пример).

в) Делимое отрицательное (***A*** < 0), делитель положительный (***B*** > 0):

1	2	3	4	5
0	$[A]^{м}_{доп}$	1 1 1 0 0 0	0 0 0 1 0	Делимое
1	$[B]^{м}_{доп}$ R'_1 $\overleftarrow{R}'_1$ $[B]^{м}_{пр}$ R_1	$\underline{0\ 0\ 0\ 0\ 0\ 0}$ 1 1 1 0 0 0 + 1 1 0 0 0 1 $\underline{0\ 0\ 1\ 1\ 1\ 1}$ $\boxed{0\ 0}$ 0 0 0 0 0 0 0 0 0 0 ЗнR1=ЗнB	$\underline{0\ 1\ 1\ 1\ 1}$ 1 0 0 0 1 0 0 0 1 | 0 0 0 0 1 | $\boxed{1}$	Сложение с делителем, выровненным по младшим разрядам Сдвиг остатка влево Сложение с делителем, выровненным по старшим разрядам Знак первого остатка не совпадает со знаком делимого – деление корректно. Формирование знака частного
2	$\overleftarrow{R}_1$ $[-B]^{м}_{доп}$ R_2	+ 0 0 0 0 0 0 $\underline{1\ 1\ 0\ 0\ 0\ 1}$ 1 1 0 0 0 1 ЗнR2≠ЗнB	0 0 1 | 1 0 0 0 1 | 1 $\boxed{0}$	Сдвиг остатка влево Вычитание делителя Формирование цифры частного
3	$\overleftarrow{R}_2$ $[B]^{м}_{пр}$ R_3	+ 1 0 0 0 1 0 $\underline{0\ 0\ 1\ 1\ 1\ 1}$ 1 1 0 0 0 1 ЗнR3≠ЗнB	0 1 | 1 0 0 0 1 | 1 0 $\boxed{0}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
4	$\overleftarrow{R}_3$ $[B]^{м}_{пр}$ R_4	+ 1 0 0 0 1 0 $\underline{0\ 0\ 1\ 1\ 1\ 1}$ 1 1 0 0 0 1 ЗнR4≠ЗнB	1 | 1 0 0 0 1 | 1 0 0 $\boxed{0}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного
5	$\overleftarrow{R}_4$ $[B]^{м}_{пр}$ R_5	+ 1 0 0 0 1 1 $\underline{0\ 0\ 1\ 1\ 1\ 1}$ 1 1 0 0 1 0 ЗнR5≠ЗнB	1 0 0 0 0 1 0 0 0 $\boxed{0}$	Сдвиг остатка влево Сложение с делителем Формирование цифры частного

В результате выполнения операции получено отрицательное частное

$[\boldsymbol{C}]_{доп} = (1.0000)_2 = (-16)_{10}$ и отрицательный остаток

$[\boldsymbol{R}]_{доп} = (1.0010)_2$, $[\boldsymbol{R}]_{пр} = (-14)_{10}$, которые соответствуют истинным значениям: $(-16) \times 15 + (-14) = -254$.

г) Оба операнда отрицательные (***A*** < 0, ***B*** < 0):

1	2	3	4	5
0	$[A]_{доп}$	1 1 0 0 0	0 0 0 1 0	Делимое
1	$[\overleftarrow{A}]_{доп}$ $[-B]_{пр}$ R_1	$+$ 1 0 0 0 0 $\underline{0\ 1\ 1\ 1\ 1}$ $\boxed{1}$ 1 1 1 1	0 0 1 0 | 0	Сдвиг делимого влево Вычитание делителя Знак первого остатка совпадает со знаком делимого – деление некорректно

ЛИТЕРАТУРА

1. Савельев А.Я. Прикладная теория цифровых автоматов. – М: Высш. шк., 1987. – 272 с.
2. Савельев А.Я. Основы информатики. – М.: Изд-во МГТУ им Н.Э. Баумана, 2001. — 328 с.
3. Угрюмов Е. Цифровая схемотехника – СПБ.: БХВ-Петербург, 2001. – 528с.
4. Вальциферов Ю.В., Дронов В.П. ИНФОРМАТИКА: ч.1. Арифметические и логические основы ЭВМ. Учебное пособие. – М: Московский государственный университет экономики, статистики и информатики: 2005. – 253 с.
5. Гуров В.В., Чуканов В.О. Основы теории и организации ЭВМ. – Интернет-университет информационных технологий - ИНТУИТ.ру, 2006.

ПРИЛОЖЕНИЕ 1

Варианты заданий

Таблица 1

№	*A*	*B*	№	*A*	*B*	№	*A*	*B*
1.	100	0,009	**41.**	850	0,105	**81.**	950	0,205
2.	200	0,008	**42.**	1050	0,305	**82.**	1150	0,405
3.	300	0,007	**43.**	1250	0,505	**83.**	1990	0,011
4.	400	0,006	**44.**	1350	0,605	**84.**	125	0,022
5.	500	0,005	**45.**	1450	0,705	**85.**	226	0,033
6.	600	0,004	**46.**	1550	0,805	**86.**	323	0,044
7.	700	0,003	**47.**	1650	0,905	**87.**	427	0,066
8.	800	0,002	**48.**	1750	0,115	**88.**	522	0,077
9.	900	0,001	**49.**	1850	0,125	**89.**	624	0,088
10.	1000	0,09	**50.**	1950	0,135	**90.**	765	0,099
11.	1100	0,08	**51.**	110	0,145	**91.**	845	0,111
12.	1200	0,07	**52.**	220	0,155	**92.**	935	0,222
13.	1300	0,06	**53.**	330	0,165	**93.**	1005	0,333
14.	1400	0,05	**54.**	440	0,175	**94.**	1125	0,444
15.	1500	0,04	**55.**	550	0,185	**95.**	1295	0,555
16.	1600	0,03	**56.**	660	0,195	**96.**	1364	0,666
17.	1700	0,02	**57.**	770	0,205	**97.**	1435	0,777
18.	1800	0,01	**58.**	880	0,215	**98.**	1524	0,888
19.	1900	0,15	**59.**	990	0,225	**99.**	1625	0,999
20.	2000	0,25	**60.**	1110	0,235	**100.**	1725	0,12
21.	150	0,35	**61.**	1220	0,245	**101.**	1825	0,23
22.	250	0,45	**62.**	1330	0,255	**102.**	1925	0,34
23.	350	0,55	**63.**	1440	0,265	**103.**	1975	0,46
24.	450	0,65	**64.**	1550	0,275	**104.**	1875	0,56
25.	550	0,75	**65.**	1660	0,285	**105.**	1775	0,67
26.	650	0,85	**66.**	1770	0,295	**106.**	1675	0,78
27.	750	0,95	**67.**	1880	0,305	**107.**	1575	0,89
28.	670	0,123	**68.**	380	0,238	**108.**	560	0,076
29.	715	0,022	**69.**	1746	0.792	**109.**	267	0.844
30.	1023	0.127	**70.**	451	0.094	**110.**	1921	0.062
31.	1140	0.794	**71.**	136	0.845	**111.**	316	0.305
32.	213	0.379	**72.**	701	0.568	**112.**	841	0.367
33.	368	0.635	**73.**	163	0.478	**113.**	372	0.631
34.	1234	0.046	**74.**	378	0.077	**114.**	1707	0.048
35.	935	0.602	**75.**	1579	0.162	**115.**	693	0.792
36.	573	0.263	**76.**	127	0.632	**116.**	1694	0.196
37.	1034	0.841	**77.**	412	0.703	**117.**	479	0.569
38.	167	0.368	**78.**	178	0.365	**118.**	589	0.946
39.	671	0.074	**79.**	1471	0.052	**119.**	1287	0.037
40.	829	0.567	**80.**	784	0.842	**120.**	703	0.479

Таблица 2

№	***R***	***S***	№	***R***	***S***
1.	41A40000	BC8A0000	**61.**	40B40000	BF3A0000
2.	C1B80000	3C4B0000	**62.**	C1C50000	3E7B0000
3.	423E5000	BD250000	**63.**	41D60000	BD6C0000
4.	C25C2000	3D1A0000	**64.**	C2E70000	3F9E0000
5.	4310DC00	BE320000	**65.**	42F8B000	BE5F0000
6.	C320FD00	3E5B0000	**66.**	C3A90700	3D4C0000
7.	419A0000	BF680000	**67.**	43BA0200	BFA40000
8.	C1AB0000	3F790000	**68.**	C0CB0000	3EB60000
9.	424B3000	BC1A0000	**69.**	40DC0000	BDF90000
10.	C23CF000	3C2D0000	**70.**	C1EDC000	3FD70000
11.	4310D500	BD460000	**71.**	41F68000	BEB30000
12.	C320E300	3D870000	**72.**	C29F4000	3DC50000
13.	C39CE000	3DB40000	**73.**	42FBD000	BFF40000
14.	42BEC000	BFBF0000	**74.**	C3FD0000	3FA10000
15.	C3CFD000	3D8B0000	**75.**	42FB0000	BED60000
16.	41C0D300	BEA80000	**76.**	C22F0000	3EA70000
17.	C3A0B000	3D7B0000	**77.**	C3AA0000	3CEC0000
18.	43F40200	BF1A0000	**78.**	42B80000	BE650000
19.	40AB0000	BEC60000	**79.**	434B0000	BEAF0000
20.	C2E48000	3DC70000	**80.**	42ED0000	BECD0000
21.	42F27000	BDE30000	**81.**	B56D0000	435CB000
22.	41FF0000	BE980000	**82.**	42AE0800	BFE70000
23.	C1E08000	3EA70000	**83.**	C3BD0500	3EC80000
24.	426A7000	BFB50000	**84.**	43CC0700	BDD90000
25.	C2A98000	3F8A0000	**85.**	C0EB0000	3FE10000
26.	43AA0000	BCDC0000	**86.**	40FA0000	BEF70000
27.	C380F000	3C4B0000	**87.**	C119C000	3D780000
28.	41FC0000	BD890000	**88.**	41288000	BFD70000
29.	C1B10000	3D470000	**89.**	C237A000	3FA50000
30.	42ABC000	BEF60000	**90.**	42429000	BDB30000
31.	C2FF0000	3EA00000	**91.**	C3598400	3FCB0000
32.	43FA0000	BFD50000	**92.**	C0D40000	3E5A0000
33.	C11F0000	3EC70000	**93.**	436A7600	BDBC0000
34.	C33A0000	3C290000	**94.**	C0780000	3F830000
35.	41B90000	BE580000	**95.**	408C0000	BEA90000
36.	43AF0000	BEBC0000	**96.**	C19DE000	3DF40000
37.	43E40000	BED00000	**97.**	41AEC000	BEDF0000
38.	B51C0000	432ABE00	**98.**	C2BFD800	3F6B0000
39.	C25A3000	3DD70000	**99.**	42D0E400	BEB90000
40.	417A7000	BE140000	**100.**	C311F200	3D570000
41.	C33B8000	3FA80000	**101.**	43230100	BF910000
42.	40A30000	BCC30000	**102.**	40E50000	BE8C0000

Таблица 2 (продолжение)

№	***R***	***S***	№	***R***	***S***
43.	C1F90000	3DFB0000	**103.**	C1F61000	3DE50000
44.	41CA0000	BD3A0000	**104.**	41E72000	BDC20000
45.	C26B6000	3E2D0000	**105.**	C2183000	3F350000
46.	42597000	BE430000	**106.**	42294200	BE050000
47.	C320A600	3FF40000	**107.**	C33A5300	3D9B0000
48.	43619800	BF160000	**108.**	434B6480	BF8C0000
49.	C8910000	3DA20000	**109.**	C03C0000	3E8A0000
50.	47AF0000	BFA10000	**110.**	41CF0000	BCD40000
51.	C2F81000	3FB50000	**111.**	C0A65000	3CE70000
52.	42BE6000	BEB20000	**112.**	43B70000	BCE70000
53.	C35A0000	3D8B0000	**113.**	C2D09000	3FB60000
54.	42F70000	BE7C0000	**114.**	43DA8000	BFA30000
55.	C1D90000	3E8E0000	**115.**	C1A78000	3E8B0000
56.	41EB0000	BF4E0000	**116.**	43CAB000	BCAF0000
57.	C1B80100	3DDE0000	**117.**	C340E700	3C870000
58.	42AED000	BFB60000	**118.**	43DE0000	BE890000
59.	C2F44000	3EC60000	**119.**	C1C47000	3FF80000
60.	43ED0000	BDC30000	**120.**	427C4B00	BEF60000

Таблица 3

№	*A*	*B*	№	*A*	*B*	№	*A*	*B*
1.	78	47	**41.**	45	81	**81.**	55	70
2.	82	46	**42.**	70	43	**82.**	63	84
3.	100	26	**43.**	63	32	**83.**	92	25
4.	73	49	**44.**	84	44	**84.**	103	27
5.	51	62	**45.**	95	33	**85.**	72	54
6.	65	53	**46.**	53	68	**86.**	90	37
7.	107	15	**47.**	45	81	**87.**	60	57
8.	67	61	**48.**	25	102	**88.**	80	23
9.	61	24	**49.**	104	21	**89.**	35	77
10.	42	80	**50.**	51	30	**90.**	83	37
11.	112	19	**51.**	72	52	**91.**	62	59
12.	38	88	**52.**	77	43	**92.**	49	63
13.	103	13	**53.**	83	19	**93.**	53	61
14.	68	55	**54.**	41	57	**94.**	37	69
15.	46	61	**55.**	101	13	**95.**	49	38
16.	81	17	**56.**	25	83	**96.**	51	29
17.	69	37	**57.**	63	33	**97.**	82	11
18.	36	79	**58.**	74	27	**98.**	56	43
19.	31	91	**59.**	81	36	**99.**	53	55
20.	109	17	**60.**	32	84	**100.**	93	32
21.	54	66	**61.**	44	83	**101.**	37	70
22.	54	67	**62.**	113	21	**102.**	64	52
23.	36	89	**63.**	38	78	**103.**	65	45
24.	51	69	**64.**	33	82	**104.**	49	58
25.	41	86	**65.**	46	68	**105.**	68	50
26.	27	93	**66.**	65	39	**106.**	73	55
27.	70	47	**67.**	67	46	**107.**	57	62
28.	85	34	**68.**	85	37	**108.**	111	18
29.	92	28	**69.**	74	48	**109.**	49	66
30.	56	61	**70.**	110	15	**110.**	92	28
31.	34	83	**71.**	11	114	**111.**	76	38
32.	67	60	**72.**	43	81	**112.**	54	72
33.	68	54	**73.**	102	25	**113.**	47	81
34.	32	95	**74.**	53	62	**114.**	35	64
35.	99	23	**75.**	79	25	**115.**	61	30
36.	47	63	**76.**	38	67	**116.**	44	39
37.	78	28	**77.**	101	17	**117.**	83	41
38.	33	85	**78.**	60	43	**118.**	49	56
39.	75	36	**79.**	39	86	**119.**	77	21
40.	48	48	**80.**	69	56	**120.**	18	70

Таблица 4

Номер варианта	*A*	*B*	Номер варианта	*A*	*B*	Номер варианта	*A*	*B*
1.	61	47	**41.**	117	14	**81.**	16	118
2.	22	81	**42.**	19	101	**82.**	38	62
3.	82	21	**43.**	119	20	**83.**	120	28
4.	20	83	**44.**	46	63	**84.**	18	121
5.	23	84	**45.**	102	17	**85.**	27	122
6.	64	37	**46.**	123	26	**86.**	24	85
7.	65	45	**47.**	67	50	**87.**	46	49
8.	66	47	**48.**	36	86	**88.**	53	15
9.	44	67	**49.**	29	124	**89.**	103	68
10.	25	87	**50.**	51	45	**90.**	35	126
11.	88	36	**51.**	104	11	**91.**	13	125
12.	89	26	**52.**	43	69	**92.**	12	22
13.	90	35	**53.**	106	21	**93.**	105	91
14.	70	34	**54.**	12	107	**94.**	33	48
15.	41	71	**55.**	52	63	**95.**	55	57
16.	72	33	**56.**	92	27	**96.**	42	109
17.	73	28	**57.**	13	108	**97.**	23	74
18.	93	25	**58.**	44	56	**98.**	41	40
19.	37	94	**59.**	14	170	**99.**	59	29
20.	95	24	**60.**	75	42	**100.**	58	111
21.	15	96	**61.**	112	11	**101.**	16	97
22.	76	32	**62.**	113	17	**102.**	32	50
23.	38	77	**63.**	62	54	**103.**	48	43
24.	78	40	**64.**	98	30	**104.**	60	115
25.	35	81	**65.**	83	27	**105.**	53	85
26.	63	77	**66.**	72	89	**106.**	38	94
27.	33	115	**67.**	76	23	**107.**	103	38
28.	62	78	**68.**	83	36	**108.**	79	49
29.	91	52	**69.**	94	18	**109.**	57	41
30.	59	68	**70.**	59	61	**110.**	48	58
31.	31	79	**71.**	114	34	**111.**	31	80
32.	18	99	**72.**	53	39	**112.**	39	41
33.	19	100	**73.**	116	30	**113.**	125	94
34.	52	61	**74.**	57	63	**114.**	24	37
35.	57	84	**75.**	35	71	**115.**	43	79
36.	70	39	**76.**	61	42	**116.**	92	38
37.	33	78	**77.**	81	59	**117.**	26	84
38.	56	66	**78.**	69	42	**118.**	58	74
39.	34	85	**79.**	76	45	**119.**	91	76
40.	20	98	**80.**	67	49	**120.**	51	63

Таблица 5

Номер варианта	*A*	*B*	Номер варианта	*A*	*B*	Номер варианта	*A*	*B*
1.	3038	31	**41.**	1571	23	**81.**	1716	26
2.	1682	24	**42.**	1536	22	**82.**	1410	15
3.	1318	19	**43.**	1303	20	**83.**	1449	21
4.	1382	18	**44.**	986	12	**84.**	1020	15
5.	1344	20	**45.**	1654	18	**85.**	1272	12
6.	1422	21	**46.**	2076	22	**86.**	1248	13
7.	1305	14	**47.**	964	12	**87.**	1152	14
8.	1630	17	**48.**	2328	27	**88.**	1904	28
9.	1834	26	**49.**	1182	12	**89.**	965	10
10.	2072	22	**50.**	1145	17	**90.**	1924	30
11.	2566	29	**51.**	1816	21	**91.**	946	10
12.	954	10	**52.**	1833	27	**92.**	1088	26
13.	982	15	**53.**	1436	19	**93.**	1380	15
14.	1916	26	**54.**	1644	23	**94.**	932	10
15.	1804	25	**55.**	981	15	**95.**	960	12
16.	1534	15	**56.**	1058	18	**96.**	1460	20
17.	1643	24	**57.**	974	11	**97.**	1080	12
18.	944	12	**58.**	1211	17	**98.**	1924	26
19.	1684	25	**59.**	1911	27	**99.**	1056	18
20.	1645	35	**60.**	1933	30	**100.**	938	14
21.	2461	31	**61.**	2164	29	**101.**	1093	27
22.	2182	27	**62.**	1375	21	**102.**	2137	22
23.	1589	24	**63.**	2194	19	**103.**	1074	11
24.	1426	19	**64.**	1054	13	**104.**	2468	26
25.	1748	18	**65.**	2389	31	**105.**	2391	28
26.	2374	27	**66.**	1987	26	**106.**	1076	13
27.	1146	17	**67.**	1654	18	**107.**	1540	22
28.	1271	18	**68.**	1022	15	**108.**	1634	19
29.	1228	17	**69.**	2076	23	**109.**	994	14
30.	1522	23	**70.**	1354	19	**110.**	924	11
31.	1435	21	**71.**	1217	18	**111.**	1440	20
32.	1036	15	**72.**	1800	23	**112.**	1056	12
33.	1302	17	**73.**	1017	13	**113.**	1462	17
34.	3184	35	**74.**	2171	23	**114.**	1584	24
35.	2781	46	**75.**	1789	26	**115.**	3012	29
36.	1734	22	**76.**	1925	28	**116.**	2345	28
37.	3267	41	**77.**	1123	14	**117.**	1894	21
38.	974	33	**78.**	968	41	**118.**	1473	15
39.	2367	25	**79.**	1885	17	**119.**	1844	18
40.	3106	39	**80.**	2043	19	**120.**	1242	27

ПРИЛОЖЕНИЕ 2

Контрольные вопросы
по теме «Представление чисел в ЭВМ»

1. Что понимается под системой счисления? (3 балла)
2. В чем отличие непозиционной системы счисления от позиционной? (3 балла)
3. Привести примеры чисел в непозиционной системе счисления, на которых пояснить, почему она так называется. (3 балла)
4. Что понимается под основанием системы счисления? (2 балла)
5. Что понимается под весом разряда? (2 балла)
6. Обосновать использование в ЭВМ двоичной системы счисления. (6 баллов)
7. Система счисления с каким основанием является оптимальной по затратам оборудования (1 балл) и почему? (5 баллов)
8. Пояснить, почему в ЭВМ для представления знаковых чисел используется дополнительный код, а не прямой? (3 балла)
9. Каким образом представляется порядок числа с плавающей запятой в процессорах Intel 80x86? (3 балла)
10. Что понимается под «скрытым» разрядом и для какой цели он используется? (4 балла)
11. Представить отрицательное число с порядком, равным $\boldsymbol{n}$, и максимальной ненормализованной мантиссой в формате Ф1. (2 балла) Определить значение этого числа. (2 балла)
12. Представить отрицательное число с порядком, равным $\boldsymbol{n}$, и максимальной мантиссой в формате Ф3. (2 балла) Определить значение этого числа. (2 балла)
13. Представить отрицательное число с минимальным порядком и максимальной мантиссой в формате Ф2. (1 балл) Определить значение этого числа. (2 балла)
14. Определить диапазон представления целых чисел без знака в формате, содержащем $\boldsymbol{m}$ двоичных разрядов ($\boldsymbol{m}$=11, 12, …). (2 балла)
15. Определить диапазон представления целых чисел со знаком в формате, содержащем $\boldsymbol{m}$ двоичных разрядов ($\boldsymbol{m}$=6, 9, …). (2 балла)
16. Определить количество двоичных разрядов, необходимых для представления целых чисел со знаком, содержащих не более $\boldsymbol{m}$ десятичных цифр ($\boldsymbol{m}$=6, 7, 8). (3 балла) Определить фактический диапазон представления целых чисел со знаком в полученном формате. (2 балла)
17. Определить количество двоичных разрядов, необходимых для представления целых чисел со знаком, входящих в диапазон $|\boldsymbol{A}| < \boldsymbol{B}$ ($\boldsymbol{B}$=3500, 7500, 9500). (3 балла) Определить фактический диапазон представления целых чисел со знаком в полученном формате. (2 балла)
18. Представить число $\boldsymbol{A}$ в форме с плавающей запятой в форматах Ф1 (4 балла), Ф2 (4 балла) и Ф3 (4 балла).

19.Определить значение числа с плавающей запятой по его шестнадцатеричному представлению ***R*** в форматах Ф1 (3 балла), Ф2 (3 балла) и Ф3 (3 балла).

20.Определить диапазон (5 баллов) и точность (2 балла) представления чисел с плавающей запятой в формате, использующем ***k*** двоичных разрядов для представления мантиссы и ***m*** разрядов для представления порядка. Порядок представляется:

а) в виде целого знакового числа в дополнительном коде;

б) в виде целого знакового числа в прямом коде;

в) в виде целого беззнакового числа со смещением, равным весу старшего разряда.

Основание порядка - ***S***.

В данном формате представить заданное число ***A***. (4 балла) Определить абсолютную и относительную погрешности представления этого числа. (3 балла) Убедиться, что относительная погрешность не превосходит точности представления чисел. (1 балл)

Возможные значения параметров:

k=4, 5, 6; ***m***=6, 8, 10, 12; ***S***=2, 4, 8, 16; ***A***=-25,1275; -0,3; 100,25; 21,0625.

21.Определить формат разрядной сетки для представления чисел с плавающей запятой из условий обеспечения заданного диапазона представления: $|A| < 10^{p}$ и заданной точности: $\sigma A_{\max} < 10^{-t}$ ($\sigma A_{\max}$ - максимальная относительная погрешность). (8 баллов) Порядок представляется:

а) в виде целого знакового числа в дополнительном коде;

б) в виде целого знакового числа в прямом коде;

в) в виде целого беззнакового числа со смещением, равным весу старшего разряда.

Основание порядка - ***S***.

Определить фактический диапазон представления чисел в полученном формате и фактическую точность их представления. (4 балла) Убедиться, что фактические значения диапазона и точности входят в заданные пределы. (2 балла)

Возможные значения параметров:

p = 50, 60, 70, 80, 90, 100; ***t*** = 8, 9, 10, 11, 12; ***S*** = 2, 4, 8, 16;

ПРИЛОЖЕНИЕ 3

Контрольные вопросы
по теме «Выполнение арифметических операций над целыми числами»

1. Поясните назначение и принцип установки арифметического флага ***OF*** (***CF, AF, SF, ZF, PF***).

2. В разрядной сетке, содержащей ***k*** (***k*** =6, 7, …) разрядов (1 – знаковый и ***k-1*** –цифровых) привести пример сложения целых чисел с разными знаками, дающий отрицательный (положительный) результат. Операнд ***A***= –17 (–27, –35, …). Операнд ***B*** и результат представить в десятичной системе счисления. Дать интерпретацию примера для беззнаковых целых чисел. Показать значения арифметических флагов.

3. В той же разрядной сетке привести пример сложения положительных (отрицательных) целых чисел, в котором результат дает переполнение разрядной сетки. Операнд ***A***=22 (– 43, 25, …). Операнд ***B*** представить в десятичной системе счисления. На примере показать возможные способы фиксации переполнения.

4. В той же разрядной сетке привести пример сложения целых чисел с одинаковыми знаками, в которых в случае положительных слагаемых возникает переполнение разрядной сетки, а в случае отрицательных слагаемых, по модулю равного положительному, переполнение не возникает. В качестве одного из слагаемых использовать число 17 (21, 39, …). Показать значения арифметических флагов.

5. В той же разрядной сетке привести пример прямого вычитания отрицательных целых чисел, при котором получаемый результат положителен (отрицателен). Операнд ***B***= –30 (вычитаемое) (***A***= –20 – уменьшаемое). Операнд ***A*** (***B***) и результат представить в десятичной системе счисления. Дать интерпретацию примера для беззнаковых целых чисел. Показать значения арифметических флагов.

6. Перечислите методы (схемы) умножения целых чисел. Какой метод (схема) умножения применяется в ЭВМ чаще и почему?

7. В чем заключается основная разница между методами умножения целых чисел в дополнительных кодах с применением и без применения коррекции?

8. В каких случаях необходима коррекция результата при умножении целых чисел в дополнительных кодах с коррекцией? В чем она заключается?

9. Чему равно максимальное значение произведения целых чисел со знаком (без знака), представленных в 6-ти (8-ми) разрядном формате? Представить это значение в разрядной сетке.

10. Выполнить операцию знакового умножения целых чисел с применением коррекции (без коррекции). ***A***=25 (множимое) и ***B***= –13 (множитель). Формат операндов выбрать самостоятельно. Проверить правильность результата.

11. При каком условии в операции знакового деления целых чисел в дополнительных кодах после выработки всех цифр частного осуществляется коррекция остатка? Как выполняется эта коррекция?

12. Привести пример фиксации некорректности деления целых чисел по методу деления в дополнительных кодах для отрицательных операндов. Формат операндов выбрать самостоятельно.

13. Выполнить операцию деления целых чисел по методу деления в дополнительных кодах. $\boldsymbol{A}= -12$ (делимое) и $\boldsymbol{B}=3$ (делитель). Формат операндов выбрать самостоятельно.

14. Привести пример фиксации некорректности деления целых чисел по методу деления в дополнительных кодах для делимого, равного –85 и максимального положительного делителя, при котором она происходит. Формат операндов выбрать самостоятельно.

С О Д Е Р Ж А Н И Е

СПбГУ ИТМО стал победителем конкурса инновационных образовательных программ вузов России на 2007–2008 годы и успешно реализовал инновационную образовательную программу «Инновационная система подготовки специалистов нового поколения в области информационных и оптических технологий», что позволило выйти на качественно новый уровень подготовки выпускников и удовлетворять возрастающий спрос на специалистов в информационной, оптической и других высокотехнологичных отраслях науки. Реализация этой программы создала основу формирования программы дальнейшего развития вуза до 2015 года, включая внедрение современной модели образования.

КАФЕДРА ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

О кафедре

Кафедра ВТ СПбГУ ИТМО создана в 1937 году и является одной из старейших и авторитетнейших научно-педагогических школ России.

Первоначально кафедра называлась кафедрой математических и счетно-решающих приборов и устройств и занималась разработкой электромеханических вычислительных устройств и приборов управления. Свое нынешнее название кафедра получила в 1963 году.

Кафедра вычислительной техники является одной из крупнейших в университете, на которой работают высококвалифицированные специалисты, в том числе 8 профессоров и 15 доцентов, обучающие около 500 студентов и 30 аспирантов.

Кафедра имеет 4 компьютерных класса, объединяющих более 70 компьютеров в локальную вычислительную сеть кафедры и обеспечивающих доступ студентов ко всем информационным ресурсам кафедры и выход в Интернет. Кроме того, на кафедре имеются учебные и научно-исследовательские лаборатории по вычислительной технике, в которых работают студенты кафедры.

Чему мы учим

Традиционно на кафедре ВТ основной упор в подготовке специалистов делается на фундаментальную базовую подготовку в рамках общепрофессиональных и специальных дисциплин, охватывающих наиболее важные разделы вычислительной техники: функциональная схемотехника и микропроцессорная техника, алгоритмизация и программирование, информационные системы и базы данных, мультимедиатехнологии, вычислительные сети и средства телекоммуникации, защита информации и информационная безопасность. В то же время, кафедра предоставляет студентам старших курсов возможность специализироваться в более узких профессиональных областях в соответствии с их интересами.

<u>Специализации на выбор</u>

Кафедра ВТ ИТМО предлагает в рамках инженерной и магистерской подготовки студентам на выбор 3 специализации.

1. Специализация в области информационно-управляющих систем направлена на подготовку специалистов, умеющих проектировать и разрабатывать управляющие системы реального времени на основе средств микропроцессорной техники. При этом студентам, обучающимся по этой специализации, предоставляется уникальная возможность участвовать в конкретных разработках реального оборудования, изучая все этапы проектирования и производства, вплоть до получения конечного продукта. Дня этого на кафедре организована специальная учебно-производственная лаборатория, оснащенная самым современным оборудованием. Следует отметить, что в последнее время, в связи с подъемом отечественной промышленности, специалисты в области разработки и проектирования информационно-управляющих систем становятся все более востребованными, причем не только в России, но и за рубежом.

2. Кафедра вычислительной техники - одна из первых, начавшая в свое время подготовку специалистов в области открытых информационно-вычислительных систем. Сегодня студентам, специализирующимся а этой области, предоставляется уникальная возможность изучать и осваивать одно из самых мощных средств создания больших информационных систем - систему управления базами данных Oracle. При этом повышенные требования, предъявляемые к вычислительным ресурсам, с помощью которых реализуются базы данных в среде Oracle, удовлетворяются за счет организации на кафедре специализированного компьютерного класса, оснащенного мощными компьютерами фирмы SUN, связанными в локальную сеть кафедры. В то же время, студенты, специализирующиеся в данной области, получают хорошую базовую подготовку в области информационных систем, что позволяет им по завершению обучения успешно разрабатывать базы данных и знаний не только в среде Oracle, но и на основе любых других систем управления базами данных.

3. И, конечно же, кафедра не могла остаться в стороне от бурного натиска вычислительных сетей и средств телекоммуникаций в сфере компьютерных технологий. Наличие высокопрофессиональных кадров в данной области и соответствующей технической базы на кафедре (две локальные вычислительные сети, объединяющие около 80 компьютеров и предоставляющие возможность работы в разных операционных средах - Windows, Unix, Solaris), позволило организовать подготовку специалистов по данному направлению, включая изучение вопросов компьютерной безопасности, администрирования, оптимизации и проектирования вычислительных сетей.

Павел Семенович Довгий
Владимир Иванович Поляков

Арифметические основы ЭВМ

Учебно-методическое пособие по выполнению домашних заданий
по дисциплине "Дискретная математика"

В авторской редакции
Дизайн В.И.Поляков
Верстка В.И.Поляков
Редакционно-издательский отдел Санкт-Петербургского государственного университета информационных технологий, механики и оптики
Зав. РИО Н.Ф. Гусарова
Лицензия ИД № 00408 от 05.11.09
Подписано к печати
Заказ №
Тираж 200 экз.
Отпечатано на ризографе

Редакционно-издательский отдел
Санкт-Петербургского государственного университета информационных технологий, механики и оптики
197101, Санкт-Петербург, Кронверкский пр., 49

ИТМО
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ