Решение задач на тему
«Представление чисел в компьютере».

Типы задач.

1. **Целые числа. Представление чисел в формате с фиксированной запятой.**
2. **Дробные числа. Представление чисел в формате с плавающей запятой.**
3. **Арифметические операции с числами в формате с плавающей запятой.**

1. Целые числа. Представление чисел в формате с фиксированной запятой.

Методические рекомендации:

В задачах такого типа используются понятия:

- Фиксированная запятая или фиксированная точка.
- Машинное слово
- Прямой код
- Дополнительный код
- Обратный код

Фиксированная запятая.

Целые числа в компьютере хранятся в памяти в формате *с **фиксированной запятой или фиксированной точкой.*** В этом случае каждому разряду ячейки памяти соответствует всегда один и тот же разряд числа, а запятая находится справа после младшего разряда, т.е. вне разрядной сетки.

Машинное слово.

Множество целых чисел, представимых в памяти ЭВМ ограничено и зависит от размера ячеек памяти **(машинного слова)**, используемых для их хранения. В k-разрядной ячейке может храниться 2^k различных значений целых чисел.

Представление целых положительных чисел.

Алгоритм№1 получения внутреннего представления <u>целого положительного числа N</u>, хранящегося в k разрядном машинном слове:

1. Перевести число N в двоичную систему счисления.
2. Полученный результат дополнить слева незначащими нулями до k разрядов

Прямой код.

Для хранения ***целых неотрицательных чисел*** отводится одна ячейка памяти **(8 бит).**

Для хранения ***целых чисел со знаком*** отводится две ячейки памяти **(16 бит),** причем старший (левый) разряд отводится под знак числа (если число положительное, то в знаковый разряд записывается 0, если число отрицательное записывается 1).

Представление в компьютере положительных чисел с использованием формата «знак-величина» называется ***прямым кодом*** **числа**.

Дополнительный код. Обратный код

Для представления отрицательных чисел используется ***дополнительный код.*** Дополнительный код позволяет заменить арифметическую операцию вычитания операцией сложения, что существенно упрощает работу процессора и увеличивает его быстродействие.

***Алгоритм №2** получения внутреннего представления целого отрицательного числа N, хранящегося в k разрядном машинном слове :*

1. Получить внутреннее представление положительного числа ***N*** (Перевести число N в двоичную систему счисления, полученный результат дополнить слева незначащими нулями до k разрядов)
2. Получить ***обратный код*** этого числа заменой 0 на 1 и 1 на 0, т.е значения всех бит инвентировать.
3. К полученному числу прибавить 1.

Данная форма называется **дополнительным кодом**

***Алгоритм №3** перевода дополнительного кода в десятичное число.*

1) Инвертировать дополнительный код
2) Прибавить к полученному коду 1 и получить модуль отрицательного числа:
3) Перевести в десятичное число и приписать знак отрицательного числа.

Уровень «3»

1. Компьютер работает только с целыми положительными числами. Каков диапазон изменения чисел, если для представления числа в памяти компьютера отводится 1 байт? ([1], стр. 135, № 46)

Решение:

Диапазон значений от 0 до 2^8-1=255

Ответ: от 0 до 255.

2. Каков диапазон изменения целых чисел (положительных и отрицательных), если в памяти компьютера для представления целого числа отводится 1 байт. ([1], стр. 135, № 47)

Решение:

В диапазоне целых положительных чисел всего 256 чисел, если в памяти компьютера для них отводится 1 байт.

Диапазон значений положительных и отрицательных чисел в равном количестве рассчитаем так: 256:2= 128. Минимальное отрицательное число равно -128. Так как число 0 также входит в этот диапазон, то максимальное положительное число будет равно 127 (от -2^{k-1} до 2^{k-1}-1, действительно, так как 2^k:2=2^{k-1}).

Ответ: от -128 до 127.

3. Пусть для представления целых чисел в компьютере используется 16 - разрядная ячейка (2 байта). Определить каков диапазон хранимых чисел, если: а) используются только положительные числа; б) используются как положительные, так и отрицательные числа в равном количестве. ([1], Пример 1, стр. 135)

Решение:

Всего в 16 – разрядной сетке может храниться 2^{16}=65536 значений. Следовательно:

а) диапазон значений только положительных чисел от 0 до 65535 (от 0 до 2^k-1, 1 отняли, так как одно значение пошло на кодировку числа 0);

б) диапазон значений положительных и отрицательных чисел в равном количестве рассчитаем так: 65536:2= 32768. Минимальное отрицательное число равно -32768. Так как число 0 также входит в этот диапазон, то максимальное положительное число будет равно 32767 (от -2^{k-1} до 2^{k-1}-1, действительно, так как 2^k:2=2^{k-1}).

Ответ: а) от 0 до 65535; б) от -32768 до 32767.

4. Заполнить таблицу, записав максимальные и минимальные значения чисел в заданном компьютерном представлении:

Компьютерное представление	*Максимальное значение*	*Минимальное значение*
целые неотрицательные числа		
целые числа со знаком		
большое целое число со знаком		

([2], стр.64, №2.52)

Решение:

Для хранения ***целых неотрицательных чисел*** отводится одна ячейка памяти **(8 бит).** Минимальное значение – все разряды заполнены 0, это будет число 0, максимальное значение – восемь единиц, или десятичное число 255.

Для хранения ***целых чисел со знаком*** отводится две ячейки памяти **(16 бит),** причем старший (левый) разряд отводится под знак числа (если число положительное, то в знаковый разряд записывается 0, если число отрицательное записывается 1). Следовательно максимальное значение целых чисел со знаком $2^{15} – 1 = 32767$ (один разряд на знак и 1 на кодирование 0), а минимальное $-2^{15} = -32768$.

Для хранения больших целых чисел со знаком отводится 4 ячейки памяти-32 бита. Значит,

максимальное значение большого целого числа со знаком $2^{31} – 1 = 2147483647$, минимальное значение $-2^{31} = -2147483648$

Ответ:

Компьютерное представление	**Максимальное значение**	**Минимальное значение**
целые неотрицательные числа	$2^8 – 1 = 255$	**0**
целые числа со знаком	$2^{15} – 1 = 32767$	$-2^{15} = -32768$
большое целое число со знаком	$2^{31} – 1 = 2147483647$	$-2^{31} = -2147483648$

Примечание, можно предложить учащимся сравнить максимальные **значения знаковых и без знаковых представлений чисел:**
Знаковое -127 и без знаковое -255 у 8 - разрядных представлений (на число 1 байт)
Знаковое - 32767 и без знаковое - 65535 у 16 - разрядных представлений (на число 2 байта).
Максимальное значение знакового числа почти в 2 раза меньше, чем у без знакового числа.
5. Компьютер работает только с целыми положительными числами. Каков диапазон изменения чисел, если для представления числа в памяти компьютера отводится 4 байта? ([1], стр.135, № 48.)

Решение:

Если компьютер работает только с целыми положительными числами, то разряд на знак выделять не надо. Диапазон чисел лежит от 0 до 2^{32} -1, так как 4 байта – 32 бит.

Ответ: от 0 до 2^{32} -1 или от 0 до 4 294 967 295

6. Каков диапазон изменения целых чисел (положительных и отрицательных), если в памяти компьютера для представления целого числа отводится 4 байта? ([1], стр.135, № 49.)

Решение:

Для хранения больших целых чисел со знаком отводится 4 ячейки памяти-32 бита. Значит, максимальное значение большого целого числа со знаком $2^{31} – 1 = 2147483647$, минимальное значение $-2^{31} = -2147483648$

Ответ: от -2147483648 до 2147483647

7. Получить внутреннее представление целого числа 1607 в 2-х байтовой ячейке. Записать ответ в 16 - ричной форме. ([1], Пример 2, стр.135.)

Решение:

Воспользуемся алгоритмом №1
$1607_{10} = 11001000111_2$

Внутреннее представление этого числа: 0000 0110 0100 0111

16- ричная форма -0647.

Ответ: 0000 0110 0100 0111 или 0647

8. Записать дополнительный код отрицательного числа –2002 для 16-ти разрядного компьютерного представления с использованием алгоритма. ([2], стр.60, пример №2.38)

Решение:

Воспользуемся алгоритмом №2

Прямой код	$\lvert -2002_{10} \rvert$	0000011111010010_2
Обратный код	инвертирование	1111100000101101_2
	прибавление единицы	1111100000101101_2 +
		0000000000000001_2
Дополнительный код		1111100000101110_2

Ответ: 1111100000101110_2

9. Заполнить таблицу, записав десятичные числа в заданном компьютерном представлении:

Десятичные числа	*Компьютерное представление*	
	целые неотрицательные числа	*целые числа со знаком*
255		
-255		
32768		
-32768		

([2], стр.64, №2.52)

Решение:

Так как для хранения ***целых неотрицательных чисел*** отводится одна ячейка памяти (8 бит), то в компьютерном представлении максимальное целое неотрицательное число это десятичное число 255. а двоичное 11111111. Значит компьютерное представление чисел, больших 255, и отрицательных, как целых неотрицательных отсутствует.

Для хранения ***целых чисел со знаком*** отводится две ячейки памяти **(16 бит),** причем старший (левый) разряд отводится под знак числа (если число положительное, то в знаковый разряд записывается 0, если число отрицательное записывается 1). Так как максимальное положительное число, которое может храниться в памяти в формате *целое число со знаком* равно 2^{15} -1 =32767, то представление числа 32768 в таком формате отсутствует. Минимальное отрицательное число, записанное в таком формате десятичное
-32768, двоичное 1000 0000 0000 0000. Число -255 представлено в дополнительном коде.

Ответ:

Десятичные числа	Компьютерное представление	
	целые неотрицательные числа	целые числа со знаком
255	11111111	0000000011111111
-255	отсутствует	1111111100000001
32768	отсутствует	отсутствует
-32768	отсутствует	1000000000000000

10. ***Выполнить арифметические действия 3 – 10 (числа записаны в 10-с.с.) в 16 разрядном компьютерном представлении.***

Решение:

+3	**+ 00000011**
-10	**11110110 – доп. код числа -10**
-7	**11111001 –доп.код числа -7**

Уровень «4»

Решение задач на основе применения определения дополнительного кода.

Опр. Дополнительный код отрицательного числа А, хранящегося в n ячейках, равен

$$2^n - |A|$$

11. Записать дополнительный код отрицательного числа -2002 для 16 –разрядного компьютерного представления.([2], стр.58, №2.37)

Решение:

Проведем вычисления в соответствии с определением дополнительного кода, где n=16:

2^{16}	=	10000000000000000_2	65536_{10}
2002_{10}	=	0000011111010010_2	2002_{10}
$2^{16} - \|2002_{10}\|$	=	1111100000101110_2	63534_{10}

Проведем проверку с использованием десятичной системы счисления. Дополнительный код 63534_{10} в сумме с модулем отрицательного числа 2002_{10} равен 65536_{10}, т.е. дополнительный код дополняет модуль отрицательного числа до 2^{16} .

Ответ: 1111100000101110_2

12. Заполнить таблицу, записав отрицательные десятичные числа в прямом, обратном и дополнительном кодах в 16-ти разрядном представлении:

Десятичные числа	*Прямой код*	*Обратный код*	*Дополнительный код*

-10			
-100			
-1000			
-10000			

([2], стр.64, №2.51)

Решение:

-10

Прямой код:

10 :2=5 (остаток **0**):2=2 (остаток **1**):2=**1** (остаток **0**)

$10_{10}=1010_2$

Прямой код 0000000000001010.

Обратный код 1111111111110101.

Дополнительный код получаем добавлением к обратному числа 1:

1111111111110110

-100

Прямой код:

$100_{10}=1*2^6+1*2^5+1*2^2=64+32+4=1100100_2$

Прямой, обратный и дополнительный код находим аналогично.

-1000

$1000_{10}=2^9+2^8+2^7+2^6+2^5+2^3=512+256+128+64+32+8=1111101000$

Прямой, обратный и дополнительный код находим аналогично.

-10000

Так как $2^{16}=65536$, а $2^{15}=32768$, $2^{14}=16384$, то в разложении числа 10000 наивысшая степень двойки число 13.

$10000_{10}=2^{13}+2^{10}+2^9+2^8+2^4=8192+1024+512+256+16=10011100010000$

Ответ:

Десятичные числа	**Прямой код**	**Обратный код**	**Дополнительный код**
-10	**0000000000001010**	**1111111111110101**	**1111111111110110**
-100	**0000000001100100**	**1111111110011011**	**1111111110011100**
-1000	**0000001111101000**	**1111110000010111**	**1111110000011000**
-10000	**0010011100010000**	**1101100011101111**	**1101100011110000**

Примечание: перевод чисел можно проделать в калькуляторе.

13. Записать в двоичной и 16-ричной форме внутреннее представление наибольшего положительного целого и наибольшего по абсолютной величине отрицательного целого числа, представленных в 1-байтовой ячейке памяти. ([1], стр.136, №50)

Решение:

1. Так как в компьютере могут быть представлены как положительные, так и отрицательные числа в однобайтовой ячейке памяти, то всего таких чисел будет 256. (2^8). Наибольшее положительное число, представленное в однобайтовой ячейке памяти (с учетом крайнего правого разряда на знак) $2^7-1=127_{10}=01111111_2=7F_{16}$

2. Наибольшее по абсолютной величине отрицательное целое число, представленное в 1-байтовой ячейке памяти число $128_{10}=1000\ 0000_2=80_{16}$

Ответ: $\mathbf{01111111_2=7F_{16}}$ **и** $\mathbf{1000\ 0000_2=80_{16}}$

14. Записать в двоичной и шестнадцатиричной форме внутреннее представление наибольшего положительного целого и наибольшего по абсолютной величине отрицательного целого числа, представленных в 2-х байтовой ячейке памяти. ([1], стр.137, №51)

Решение:

1. Так как в компьютере могут быть представлены как положительные, так и отрицательные числа в ***2-х байтовой*** ячейке памяти, то всего таких чисел будет 2^{16}.

Наибольшее положительное число, представленное в двухбайтовой ячейке памяти (с учетом крайнего правого разряда на знак) $2^{15}-1 = 32767_{10} = 0\ 111111111111111_2 = 7FFF_{16}$

2. Наибольшее по абсолютной величине отрицательное целое число, представленное в 2-байтовой ячейке памяти является минимальным отрицательным числом, записанным в таком формате: $-32768_{10} = 1000\ 0000\ 0000\ 0000_2 = 8000_{16}$

Ответ: $7FFF_{16}$, 8000_{16}

15. ***Получить десятичное представление числа по его дополнительному коду 10010111_2***

Решение:

1.) Инвертируем дополнительный код 10010111_2.

Получим 01101000 – обратный код

2) Прибавим к полученному числу 1. Получим число 01101001

3) Переведем полученную запись числа из двоичной в 10-ю форму. Получим число 105.

4) Перед полученным числом поставим знак «-»

Ответ: -105

16. Получить дополнительный код десятичного числа – 105.

Решение:

1)Модуль числа записать в прямом коде в n двоичных разрядах.

$105 = 01101001_2$

2) Получить обратный код числа. Получим 10010110

3) К полученному обратному коду прибавить 1. Получим 10010111

Ответ: дополнительный код числа –105 равен 10010111

Уровень «5»

Используются алгоритмы №1, 2, 3.

17. Выполнить арифметическое действие 3000_{10} - 5000_{10} в 16-ти разрядном компьютерном представлении. ([2], стр.61, №2.40)

Решение:

Представим положительное число в прямом, а отрицательное число в дополнительном коде:

Десятичное число	Прямой код	Обратный код	Дополнительный код
3000	0000101110111000		
-5000	0001001110001000	1110110001110111	1110110001110111 +0000000000000001 1110110001111000

Сложим прямой код положительного числа с дополнительным кодом отрицательного числа. Получим результат в дополнительном коде:

3000-5000			1111100000110000

Переведем полученный дополнительный код в десятичное число, воспользуемся алгоритмом №3:

1) Инвертируем дополнительный код: 0000011111001111
2) Прибавим к полученному коду 1 и получим модуль отрицательного числа:

$$\begin{array}{r} 0000011111001111 \\ +\ 0000000000000001 \\ \hline 0000011111010000 \end{array}$$

3) Переведем в десятичное число и припишем знак отрицательного числа: -2000.

Ответ: **0000011111010000**

18. Назовите достоинства и недостатки представления чисел в формате с фиксированной запятой.

Решение:

Достоинства:

- Простота
- Наглядность представления чисел
- Благодаря использованию дополнительного кода вычитание сводится к сложению, что упрощает алгоритм реализации арифметических операций.

Недостатки:

Конечный диапазон представления величин недостаточен для решения математических, физических, экономических и других задач, где используются очень малые и очень большие числа.

19. Выполнить арифметическое действие 20_{10} - 60_{10} в 16-ти разрядном компьютерном представлении. ([2], стр.64, №2.54)

Решение:

1. Представим положительное число в прямом, а отрицательное число в дополнительном коде:

Десятичное число	Прямой код	Обратный код	Дополнительный код
20	0000000000010100		
-60	0000000000111100	1111111111000011	$\begin{array}{r} 1111111111000011 \\ +\ 0000000000000001 \\ \hline 1111111111000100 \end{array}$

2. Сложим прямой код положительного числа с дополнительным кодом отрицательного числа. Получим результат в дополнительном коде:

20-60			1111111111011000

3. **Проверка:** Переведем полученный **дополнительный код в десятичное** число:
1) Инвертируем дополнительный код: 0000000000100111
2) Прибавим к полученному коду 1 и получим модуль отрицательного числа:

$$\begin{array}{r} 0000000000100111 \\ +\ 0000000000000001 \\ \hline 0000000000101000 \end{array}$$

3) Переведем в десятичное число $101000_2 = 2^5 + 2^3 = 32+8 = 40_{10}$ и припишем знак отрицательного числа: - 40. Действительно: 20-60 = -40

Ответ: 1111111111011000

2. Дробные числа. Представление чисел в формате с плавающей запятой.

Методические рекомендации:

В задачах такого типа используются понятия:

- **Плавающая запятая или точка**
- **Экспоненциальная форма числа**
- **Мантисса**
- **Порядок числа**
- **Нормализованная форма числа**
- **Обычная точность**
- **Двойная точность**

Плавающей запятой или плавающая точка - положение запятой в записи числа может изменяться.
Пример: $555{,}55 = 55555 \cdot 10^{-2} = 0{,}55555 \cdot 10^{3}$

Любое число A может быть представлено в **экспоненциальной форме**:

$$A = m \cdot q^{n}$$ **, где**

m – мантисса числа, **q** – основание системы счисления., **n – порядок числа.**

Пример: $0{,}55555 \cdot 10^{3}$
Нормализованная форма числа.
Чтобы привести к какому-то стандарту в представлении чисел с плавающей запятой условились представлять числа в **нормализованной форме.**
При этом мантисса отвечает условию: она должна быть правильной дробью и иметь после запятой цифру, отличную от нуля.

$$1/n \le |m| < 1$$

Пример:555,55 – естественная форма

$0{,}55555 \cdot 10^{3}$ - нормализованная форма
$0{,}55555 > 1/3 \approx 0{,}3333\ldots$
$0{,}55555 < 1$
Это касается и отрицательных чисел, т.к. мантисса в условии взята **по модулю**.
Дробные числа занимают в памяти 4 байта **(обычная точность)** или 8 байтов **(двойная точность).**
Для записи таких чисел выделяются разряды для хранения:

- знака мантиссы,
- знака порядка,
- порядка числа
- мантиссы.

1-й байт	***2-й байт***	***3-й байт***	***4-й байт***
± порядок	ЗНАК И **М А Н**	**Т И С**	**С А**

•в старшем бите 1-го байта хранится знак порядка числа

0 – обозначает плюс, 1 – минус;

- 7 бит 1 байта содержат **порядок;**
- в следующих трех байтах, хранятся **значащие цифры мантиссы и её знака** (24 разряда).

Уровень «3»

20. Для представления вещественного числа отводится 8 байт. Порядок занимает 11 бит. Сколько значащих цифр будет содержать двоичная мантисса? ([1], стр.140, №54)

Решение:

Число занимает 64 разряда, из них 11 разрядов на машинный порядок, значит, на знак числа и мантиссу отводится 64-11 =53 бит, на мантиссу 52 бита

Ответ: 52 бита.

21. Записать следующие числа в форме с плавающей запятой и нормализованной мантиссой: а) 217, 934_{10}; б) 75321_{10}; в) $10{,}0101_{10}$; г) 200450_{10} ([2], стр.64, №2.55)

Решение:

а) 217, 934_{10} =0,217934•10^3 , где 0,217934 –нормализованная мантисса, порядок -3
б) 75321_{10} =0,75321•10^5 , где 0,75321 –нормализованная мантисса, порядок -5
в) $10{,}0101_{10}$ = 0, 100101•10^2 , где 0, 100101 –нормализованная мантисса, порядок -2
г) 200450_{10} = 0,200450 •10^6 , где 0,200450 –нормализованная мантисса, порядок -6

22. Приведенные ниже числа распределите в два столбика: в первый поместите числа в естественной форме представления, во второй — в экспоненциальной. ([2], стр.?, №2.68)

0,1236, 123,6258; 123628 $\times 10^5$; -12,365 $\times 10^{-9}$; 0,110011 $\times 2^{100}$;
1,000001; -1111111; 1111111 $\times 2^{-11}$; 9999,9999; -1221 $\times 10^{-5}$

Решение:

Числа в естественной форме	Числа в экспоненциальной форме
0,1236	-1221 $\times 10^{-5}$
123,6258	123628 $\times 10^5$
1,000001	-12,365 $\times 10^{-9}$
-1111111	0,110011 $\times 2^{100}$
9999,9999	1111111 $\times 2^{-11}$

23. Запишите число 2001,2001 пятью различными способами в форме с плавающей запятой. ([2], стр.?, №2.69)

Решение:

Возможны такие варианты записи:
2,0012001 $\times 10^3$;
20,012001 $\times 10^2$;
200,12001 $\times 10^1$;
20012,001 $\times 10^{-1}$;
200120,01 $\times 10^{-2}$.

24. Запишите следующие числа в естественной форме:

а) 0,380456 $\times 10^2$;   в) .1100000E-5;
б) 0,200000 $\times 10^{-5}$;   г) .7892101E+5.

([2], стр.?, №2.71)

Решение:

а) 38,0456; б) 0,000002; в) 0,0000011; г) 78921,01.

Уровень «4»

25. Сравните следующие числа:

а) $318{,}4785 \times 10^{9}$ и $3{,}184785 \times 10^{11}$;

б) $218{,}4785 \times 10^{-3}$ и $1847{,}85 \times 10^{-4}$;

в) $0{,}1101 \times 2^{10}$ и 101×2^{-11};

г) 11011×2^{-100} и $1{,}1101 \times 10^{-1}$.

([2], стр.?, №2.72)

Решение:

а) $318{,}4785 \times 10^{9} = 3{,}184785 \times 10^{11}$;

б) $218{,}4785 \times 10^{-3} > 1847{,}85 \times 10^{-4}$;

в) $0{,}1101 \times 2^{10} > 101 \times 2^{-11}$; ($11{,}01 > 0{,}101$, т.к. $10_2 = 2_{10}$; $-11_2 = -3_{10}$)

г) $11011 \times 2^{-100} > 1{,}1101 \times 10^{-1}$. ($1{,}1011 > 0{,}11101$, т.к. $-100_2 = -4_{10}$)

Уровень «5»

Для решения задач можно использовать инженерный калькулятор.
Использовать дополнительный теоретический материал (см. [1], стр.138)
Рассматриваются понятия:

- **Машинный порядок.**

Для примера рассмотрим внутреннее представление вещественного числа в 4-х байтовой ячейке памяти.

В ячейке должна содержаться следующая информация о числе: знак числа, порядок и значащие цифры мантиссы.

± маш.порядок	М А	Н Т И С	С А
1-й байт	2-й байт	3-й байт	4-й байт

В старшем бите 1-го байта хранится знак числа: 0 обозначает плюс, 1 — минус. Оставшиеся 7 бит первого байта содержат **машинный порядок.** В следующих трех байтах хранятся значащие цифры мантиссы (24 разряда).

В семи двоичных разрядах помещаются двоичные числа в диапазоне от 0000000 до 1111111. Значит, **машинный порядок** изменяется в диапазоне от 0 до 127 (в десятичной системе счисления). Всего 128 значений. **Порядок** (в математическом понимании), очевидно, может быть как положительным так и отрицательным. Разумно эти 128 значений разделить поровну между положительными и отрицательными значениями порядка: от -64 до 63.

Машинный порядок смещен относительно математического и имеет только положительные значения. Смещение выбирается так, чтобы минимальному математическому значению порядка соответствовал нуль.

Связь между машинным порядком **(Mp)** и математическим **(p)** в рассматриваемом случае выражается формулой: **$Mp = p + 64$.**

Полученная формула записана в десятичной системе. В двоичной системе формула имеет вид: **$Mp_2 = p_2 + 100\ 0000_2$.**

- **Внутреннее представление вещественного числа.**

Алгоритм записи **внутреннего представления вещественного числа**:

1) перевести модуль данного числа в двоичную систему счисления с 24 значащими цифрами;

2) нормализовать двоичное число;
3) найти машинный порядок в двоичной системе счисления;
4) учитывая знак числа, выписать его представление в 4-х байтовом машинном слове.

- **Формула для вычисления количества вещественных чисел**, точно представимых в памяти компьютера: $\mathbf{N = 2^t \times (U - L + 1) + 1}$.

Здесь **t** — количество двоичных разрядов мантиссы; **U** — максимальное значение математического порядка; **L** — минимальное значение порядка.
Для рассмотренного нами варианта **(t = 24, U = 63, L = - 64).** Получается количество вещественных чисел, представимых в памяти компьютера с обычной точностью
N = 2 146 683 548.

26. Определить максимальное число и его точность для формата чисел обычной точности, если для хранения порядка и его знака отводится 8 разрядов, а для хранения мантиссы и ее знака 24 разряда. ([2], стр.62,* Пример 2.42*)

Решение:

Для формата чисел обычной точности выделяется 4 байт или 32 бит.

0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
знак и порядок								знак и мантисса																							

Максимальное значение порядка числа составит $1111111_2 = 127_{10}$, и, следовательно, максимальное значение числа составит:

$$\mathbf{2^{127} = 1{,}7014118346046923173168730371588 \times 10^{38}}$$

Точность вычислений определяется количеством разрядов, отведенных для хранения мантиссы чисел. 32-8=24 бит на знак мантиссы и мантиссу. Максимальное значение положительной мантиссы равно:

$$\mathbf{2^{23} - 1 \approx 2^{23} = 2^{(10 \times 2{,}3)} \approx 1000^{2{,}3} = 10^{(3 \times 2{,}3)} \approx 10^{7}}$$

Таким образом, максимальное значение *чисел обычной точности* с учетом возможной точности вычислений составит $1{,}701411 \times 10^{38}$ (количество значащих цифр десятичного числа в данном случае ограничено 7 разрядами).

Ответ: $1{,}701411 \times 10^{38}$

27. Определите максимальное число и его точность для формата чисел двойной точности, если для хранения порядка и его знака отводится 11 разрядов, а для хранения мантиссы и ее знака 53 разряда. ([2], стр.64, №2.56)

Решение:

Так как на хранение порядка и его знака отводится 11 разрядов, то на один порядок отводится 10 разрядов. Тогда максимальное значение порядка $1111111111_2 = 1023_{10}$
Следовательно, максимальное значение числа составит $2^{1023} =$
$8{,}9884656743115795386465259539451 \cdot 10^{307}$

Точность вычислений определяется количеством разрядов, отведенных для хранения мантиссы чисел. Так как для хранения мантиссы и ее знака отводится 53 разряда, то на саму мантиссу отводится 52 знака и максимальное значение положительной мантиссы равно
$2^{52} - 1 \approx 2^{52} = 2^{(10*5.2)} \approx (2^{10})^{5,2} \approx 1000^{5,2} = 10^{(3*5,2)} = 10^{15,6}$
Максимальное значение числа двойной точности с учетом возможной точности вычислений составит
$8{,}98846567431157 * 10^{307}$
(количество значащих цифр десятичного числа в данном случае ограничено 15-16 разрядами).
Ответ: $8{,}98846567431157 * 10^{307}$

28. Записать внутреннее представление числа 250,1875 в форме с плавающей точкой. ([1], стр.139, пример №4)

Решение.

1. Переведем его в двоичную систему счисления с 24 значащими цифрами:

$$250{,}1875_{10} = 11111010,\ 0011000000000000_2.$$

2. Запишем в форме нормализованного двоичного числа с плавающей точкой:

$$0{,}111110100011000000000000 \quad x \quad 10_2^{1000}.$$

Здесь мантисса, основание системы счисления ($2_{10} = 10_2$) и порядок ($8_{10} = 1000_2$) записаны в двоичной системе.

3. Вычислим машинный порядок в двоичной системе счисления: $\mathbf{Mp_2 = p_2 + 100\ 0000_2}$.

$$Mp_2 = 1000 + 100\ 0000 = 100\ 1000.$$

4. Запишем представление числа в 4-х байтовой ячейке памяти с учетом знака числа:

0	1001000	11111010	00110000	00000000
31	24	23		0

Шестнадцатеричная форма: **48FA3000.**

Ответ: внутреннее представление числа *250,1875 равно*
01001000 11111010 00110000 00000000
Шестнадцатеричная форма: 48FA3000.

29. По шестнадцатеричной форме внутреннего представления числа в форме с плавающей точкой C9811000 восстановить само число. ([1], стр.139, пример №5)

Решение.

1. Перейдем к двоичному представлению числа в 4-х байтовой ячейке, заменив каждую шестнадцатеричную цифру 4-мя двоичными цифрами:
1100 1001 1000 0001 0001 0000 0000 0000

1	100 1001	1000 0001	0001 0000	0000 0000
31	$\mathbf{Mp_2}$	23		0

2. Заметим, что получен код отрицательного числа, поскольку в старшем разряде с номером 31 записана 1. Получим порядок числа из уравнения: $\mathbf{Mp_2 = p_2 + 100\ 0000_2}$;
$\mathbf{p_2} = 1001001_2 – 100\ 0000_2 = 1001_2 = 9_{10}$.
3. Запишем в форме нормализованного двоичного числа с плавающей точкой с учетом знака числа: -0,1000 0001 0001 0000 0000 0000 x 2^{1001}.
4. Число в двоичной системе счисления имеет вид: $-100000010{,}001_2$.
5. Переведем число в десятичную систему счисления:
$-100000010{,}001_2 = -(1 \times 2^8 + 1 \times 2^1 + 1 \times 2^{-3}) = -258{,}125_{10}$.

30. Для представления вещественного числа отводится 2 байта. Порядок занимает 7 бит. Сколько различных вещественных чисел точно представимы в памяти такого компьютера? ([1], стр.140, №53)

Решение:

1. Используем формулу для вычисления количества вещественных чисел, точно представимых в памяти компьютера: $\mathbf{N = 2^t \times (U - L + 1) + 1}$.
Здесь **t** — количество двоичных разрядов мантиссы; **U** — максимальное значение математического порядка; **L** — минимальное значение порядка.
t=9 (16 разрядов всего, 7-машинный порядок, 16-7=9)

2. Так как машинный порядок 7 бит, 1 разряд на знак порядка, 6 бит на число порядка. Машинный порядок изменяется в диапазоне от 0 до 63 (всего значений 2^6 =64). Минимальное значение порядка **L= -32**, максимальное значение порядка **U = 31.**
3. Подставляем найденные значения в формулу:
$N = 2^t$ x (U - L + 1) + 1.
$N = 2^9$ x (31 + 32 + 1) + 1 =512*64 +1=32769

Ответ: 32769

31. Минимальное значение математического порядка в десятичной системе счисления равно (-1024). Чему равно смещение? ([1], стр.140, №55)

Решение:

Машинный порядок смещен относительно математического и имеет только положительные значения. **Смещение выбирается так, чтобы минимальному математическому значению порядка соответствовал нуль.**
Связь между машинным порядком **(Mp)** и математическим **(p)** в рассматриваемом случае выражается формулой: **Mp = p + 64, где 64** - смещение для представления в 64 байтовой ячейке памяти.
Если представить это на шкале, то имеем:
$-64 \longleftarrow \quad 0 \quad \longrightarrow \quad 63$

В данной задаче Минимальное значение математического порядка в десятичной системе счисления равно (-1024).
На шкале это можно представить так:
$-1024 \longleftarrow \quad 0 \quad \longrightarrow \quad 1023$
Легко видеть, смещение равно 1024.

Ответ: 1024.

32. Получить шестнадцатеричную форму внутреннего представления отрицательного числа -123,125 в формате с плавающей точкой в 4-х байтовой ячейке. ? ([1], стр.140, №55)

Решение:

Используем алгоритм записи **внутреннего представления вещественного числа**:

1. Переведем модуль числа в двоичную систему счисления с 24 значащими цифрами.
123_{10} =1111011_2 $0{,}125_{10}$ =$0{,}\ 001_2$
$123{,}125_{10}$ =$1111011{,}\ 00100000000000000_2$ (4 байта-32 разряда, 1 байт на знак и порядок, 3 байта или 24 бита на мантиссу)
2. Запишем в форме нормализованного двоичного числа с плавающей запятой:
0, 111101100100000000000000 x $10_2{}^{111}$ ($111_2 = 7^{10}$)
3. Вычислим машинный порядок в двоичной системе счисления.
$Mp_2 = p_2 + 100\ 0000_2 = 111_2 + 100\ 0000_2 = 1000111_2$
4. Запишем представление числа в 4-х байтовой ячейке памяти с учетом знака числа:

1	1000111	1111 0110	0100 0000	0000 0000
31	24	23		0

Шестнадцатеричная форма: **- 47F64000.**
Ответ: - 47F64000

33. *Для представления вещественного числа используется 2-х байтовая ячейка памяти. В 1-ом байте содержится знак числа и порядок, во 2-ом байте — мантисса. Определить минимальное и максимальное по абсолютной величине числа, точно представимые в таком компьютере.*

№ 58

В «игрушечном» компьютере для представления вещественных чисел используется однобайтовая ячейка памяти (биты нумеруются от 0 до 7 справа налево). 7-й бит — знак числа; 5 и 6 биты — машинный порядок; 4 — 0 биты — мантисса. Определить: 1) количество точно представимых вещественных чисел; 2) 7 наименьших десятичных чисел, представимых точно в таком компьютере.

№ 59

Говорят, что число, превышающее максимальное значение, представимое в компьютере, вызывает переполнение. Определить для «игрушечного» компьютера (задача № 58), какие из следующих чисел вызовут переполнение: 0,5; 10,0; 4,3; 8,1; 7,8.

№ 60

«Игрушечный» компьютер сохраняет значение числа, не вызывающего переполнение и не представленного точно, в виде ближайшего снизу (по абсолютной величине) точно представимого числа. Какие значения примут следующие числа в таком компьютере: 1,25; 1,6; 1,9?

№ 61

Увидит ли разницу «игрушечный» компьютер между следующими парами чисел: 1) 1,4 и 1,5; 2) 1,6 и 1,62; 3) 1,8 и 1,9?

3.Арифметические операции с числами в формате с плавающей запятой.

Методические рекомендации:

При решении задач учащиеся используют:

- **Алгоритм сложения и вычитания чисел в формате с плавающей запятой:**
  1. Провести выравнивание порядков
  2. Сложить или вычесть мантиссы.
  3. Привести полученное число к стандартному формату с плавающей запятой путем нормализации.

Процедура выравнивания порядков: порядок меньшего (по модулю) числа увеличивается до величины порядка большего (по модулю) числа. Чтобы величина числа не изменилась, мантисса уменьшается в такое же количество раз (сдвигается в ячейке памяти вправо на количество разрядов, равное разности порядков чисел).

Процедура нормализации: сдвиг мантиссы влево или вправо так, чтобы ее первая значащая цифра попала в первый разряд после запятой.

- **Алгоритм умножения чисел в формате с плавающей запятой:**
  1. Сложить порядки
  2. Перемножить мантиссы

- **Алгоритм деления чисел в формате с плавающей запятой:**
  1. Из порядка делимого вычесть порядок делителя
  2. Мантиссу делимого делить на мантиссу делителя.

Уровень «3»

31. Произвести сложение чисел $0,1\times2^3$ и $0,1\times2^5$ в формате с плавающей запятой. ([2], стр.63, №2.43)

Решение:

Произведем выравнивание порядков и сложение мантисс:

$\boldsymbol{0,1\times2^{3} = X\times2^{5}, X=(0,1\times2^{3})/ 2^{5}=0,1\times2^{-2}=0,001}$

$0,001\times2^{5}$

$$\begin{array}{r} {}^{+}0{,}100\times 2^{5} \\ \hline 0{,}101\times 2^{5} \end{array}$$

Ответ: $0{,}101\times 2^{5}$

32. ***Произвести умножение чисел $0{,}1\times 2^{3}$ и $0{,}1\times 2^{5}$ в формате с плавающей запятой. ([2], стр.63, №2.44)***

Решение:

После умножения будет получено число $0{,}01\times 2^{8}$, которое после нормализации примет вид $0{,}1\times 2^{7}$.

Ответ: $0{,}1\times 2^{7}$.

Уровень «4»

33. Произвести сложение, вычитание, умножение и деление чисел $0{,}1\times 2^{2}$ и $0{,}1\times 2^{-2}$ в формате с плавающей запятой. ([2], стр.64, №2.57)

Решение:

Произведем выравнивание порядков и сложение мантисс:

$0{,}1\times 2^{-2} = X \times 2^{2}$, $\quad X = (0{,}1\times 2^{-2})/ 2^{2} = 0{,}1\times 2^{-4} = 0{,}00001$

$$\begin{array}{r} 0{,}10000\times 2^{2} \\ {}^{+}0{,}00001\times 2^{2} \\ \hline 0{,}10001\times 2^{5} \end{array}$$

Произведем вычитание мантисс и процедуру нормализации:

$$\begin{array}{r} 0{,}10000\times 2^{2} \\ {}^{-}0{,}00001\times 2^{2} \\ \hline 0{,}01111\times 2^{2} \end{array} = 0{,}1111\times 2^{1}$$

Используем алгоритм умножения: сложим порядки и перемножим мантиссы.

$$\begin{array}{r} 0{,}10000\times 2^{2} \\ {}^{x}0{,}00001\times 2^{2} \\ \hline 0{,}000001\times 2^{4} \end{array}$$, нормализуем ответ $0{,}1\times 2^{-1}$

Используем алгоритм деления чисел в формате с плавающей запятой: из порядка делимого вычесть порядок делителя, мантиссу делимого делить на мантиссу делителя.

$$\begin{array}{r} 0{,}10000\times 2^{2} \\ {}^{:}0{,}00001\times 2^{2} \\ \hline 10\,000\times 2^{0} \end{array}$$, нормализуем ответ $0{,}1\times 2^{5}$

Ответ: $0{,}10001\times 2^{5}$; $0{,}1111\times 2^{1}$; $0{,}1\times 2^{-1}$; $0{,}1\times 2^{5}$

Литература:

[1] Информатика. Задачник-практикум в 2 т. /Под ред. И.Г. Семакина, Е.К. Хеннера: Том 1. – Лаборатория Базовых Знаний, 1999 г. – 304 с.: ил.

[2] Практикум по информатике и информационным технологиям. Учебное пособие для общеобразовательных учреждений / Н.Д. Угринович, Л.Л. Босова, Н.И. Михайлова. – М.: Бином. Лаборатория Знаний, 2002. 400 с.: ил.