コンピュータ

第 4 章の演習問題

氏名　　：津波古正輝

学籍番号：e075739A

提出期限：7 月 23 日

所属　　：情報工学科

演習問題：

4.1 コンピュータのデータ表現に関する次の説明を読んで、設問中(　)に入れるべき適当な数値を解答群の中から選べ。なお、解答は重複して選んでもよい

[データ表現に関する説明]

(1) データは全て 8 ビットで表現する。

(2) 数値は 8 ビットの 2 進数で表現する。

(3) 設問中、算術データとは、第 0 ビットを符号とする 2 進数とする。また論理データとは、8 ビットのビットパターンを符号なしの 2 進数と見なした者とする。シフト演算もこれに準ずる。

(4) 論理右シフト演算を行った時、空いたビット位置には全て 0 が入るが、算術右シフトの場合は、符号ビットと同じものが入る。

(5) 算術シフトでは、符号ビットは元のまま不変であるが、論理左シフトでは 8 ビット全てがシフトの対象となる。いずれの場合も空いたビット位置には 0 がはいる。

(6) 図は 1 ビット算術右シフトを行った例であり、図 2 は 1 ビット算術左シフトを行った例である。

[設問１]

　このコンピュータでは、算術データでは(a)の範囲の値を、論理データは(b)の範囲の値を表現できる。

答え

a=エ　b=ア

解説：

8 ビットで表現するので、2 の 8 乗=0〜255(256 個)表現できるのだが、1 ビット、符号に使用するので、2 の 7 乗=0〜127(128 個)の表現となる。これは、正の正数だけなので、負の整数も入れると、-127〜0 もはいるので、-127〜127 までの表現が可能。よって、-128 より大きく、128 より小さい値の表現ができる。

データ表現に関する説明で、算術データと論理データの定義がされているので、よって、答えは、a=エ、b=アとなる。

[設問２]

　１０進数の－５を２進数で表現すると、(c)となる。これは、論理データと見なした場合、１０進数で表現すると(d)である。

答え

c=ウ　d=カ

解説：

正の正数『5』を 2 進数で表すと、00000101 となる。負の正数『-5』を求める為には、5 の補数を求めると良い。つまり、『00000101』のすべてのビットを反転させて、最後に最下位ビットに +1 をすれば良い。よって、-5=111111011 となる。また、11111011 を 10 進数になおすと、128+64+32+16+8+2+1=251 となる。よって、答えは c=ウ d=カとなる。

[設問３]

　10 進数の－５を表現する２進数を左へ３ビットシフトした結果を２進数で表現すると、算術シフトでは(e)、論理シフトでは(f)となる。また、算術シフトでは、溢れがなければ(g)倍することに等しい。

答え

e=ア　f=ア　g=ウ

解説：

これはただ、左に三個分ずらせば良い。注意するのが、データ表現に関する説明でもあるように、『論理右シフト演算を行った時、空いたビット位置には全て０が入るが、算術右シフトの場合は、符号ビットと同じものが入る』ということを気おつける。（この問題では、もともと１だったので大丈夫)。

そして、左に３個分シフトするということは、×８倍するということだ。

例；0001→1000(2 進数)とすると、1→8(10 進数)となる。

よって、答えは、e=ア　f=ア　g=ウとなる。

［設問４］

　一般に正数データを左へｎビットシフトすることは(h)倍にすることに等しく、右へｎビットシフトすることは(i)倍することに等しい。ただし、右へｎビットシフトした結果、あふれ出たビットの中に１が１つでもあると、その分は(j)となる。

h=ア　i=カ　j=ア

解説：

上の解説で述べた例を使うと、3 個分ずらして×８倍なので、8=2 の 3 乗というのがわかる。よって、n 個ずらす(シフト)することは、2 の n 乗倍することと同じである。

また、左にシフトすることが、乗算ということは、右にシフトすることが、除算であることが予想でき、実験をしてみると、

実験(2 個左にシフト)；1100→0011(2 進数)とすると 12→3(10 進数)より、4 で割られているのがり、上の予想が当っているのが分かる。よって、答えは、h=ア　i=カ　j=ア

[a、b に関する解答群]

ア　０～２５５　　　　イ　０～５１１　　　　ウ　０～１０２３

エ　－１２８～１２８　　オ　－２５６～２５５　　カ　－５１２～５１１

[c、e、f に関する解答群]

ア　11011000　　イ　00001101　　ウ　11111011　　エ　01110011

オ　00000101　　カ　10000101　　キ　11111010

[d、g]

ア　２　　イ　４　　ウ　８　　エ　１６　　オ　３２

カ　２５１　キ　２５２　ク　２５３　ケ　２５４　コ２５５

[h、i に関する解答群]

ア 2のn乗　　イ 4のn乗　　ウ 8のn乗　　エ 16のn乗　　オ 32のn乗

カ 1/2のn乗　キ1/4のn乗　ク1/8のn乗　ケ1/16のn乗 コ1/32のn乗

[jに関する解答群]

ア　切捨て　　イ　切上げ　　ウ　四捨五入　　エ　不定

4.2 論理演算に関する設問に答えよ。

[設問]

入力変数A,Bに対する出力関数Fが次の真理値表に示すものであるとき、(1),(5)の各出力関数を得るに適する論理式a～e、およびその名称f～jを解答群の中から選べ。

なお、論理式中の”,”は論理積を、”+”は論理和を、”X”はXの否定を表すものとする。

入力変数	A	0011		
出力変数	B	0101	論理式	名称
(1)	F	0001	a	f
(2)	F	1000	b	g
(3)	F	1110	c	h
(4)	F	0110	d	i
(5)	F	1001	e	j

a=イ　b=オ　c=ア　d=ウ　e=エ　f=イ　g=エ　h=ア　i=オ　j=ウ

解説：

まず、解答群の式の計算をして、見比べていく。

Aの否定=1100　　　　　　　Bの否定=1010

ア：0011と0101のANDの否定=0001の否定=1110

イ：0011と0101のAND=0001

ウ：0011と1010のAND+1100と0101のAND=0010と0100のOR=0110

エ：0011と0101のAND+1100と1010のAND=0001と1000のOR=1001

オ：0011と0101のORの否定=0111の否定=1000

あとは、これを見比べて言葉(論理式)を考えれば良い。論理名の判別は、AとBとFを見比べる。

[a～e に関する解答群]

ア　F=A・B　　　イ　F=A・B　　　ウ　F=A・B+A・B

エ　F=A・B+A・B　　　　　　　オ　A+B

[f～j に関する解答群]

ア　否定論理式（NAND 演算）　　　イ　論理積（AND 演算）

ウ　一致演算（投下演算）　　　　　エ　否定論理和（NOR 演算）

オ　排他的論理和（EOR 演算）

4.3 次の記述による前提を与えた時、その結論として正しいものを解答群の中から選べ。

(a)命題 A,B がある。A ならば B である。

(b)集合 A,B,C がある。A は B を含み B は C を含まない。

(c)論理値 AB があり、A は真、B は偽である。

a=エ　b=ウ　c=エ

解説：

(a)：A ならば B である。これは、例えると分かりやすい。A=『ケーキ』、B=『うまい』と例えよう。

『うまい』なら、『ケーキ』なのか？　違う。野菜炒め、すき焼き等たくさんある。よって、アは違う。

『うまい』ならば『ケーキ』でないのか？　違う。うまいケーキは世の中に沢山ある。よって、イは違う。

『うまくない』ならば『ケーキ』なのか？世の中には、うまくないものは沢山？ある。よって、ケーキだけとは限らない。よって、ウは違う。

『うまくない』ならば、『ケーキでない』なのか？最初にうまくない、といっている。上でも述べたように、うまくないものは沢山あり、ケーキでない可能性もある。よって、これが正解。

(b)：これは、図を書くと分かる。

よって、ウが正解

(c)：これは、真理値表を書くとわかる。

よって、正解はエ

[(a)に関する解答群]

ア　B ならば A である　　　　　　　　イ　B ならば A でない

ウ　B でなければ A である　　　　　　エ　B でなければ A でない

[(b)に関する解答群]

ア　A は C を含む　　　　　　　　　　イ　A は C を含まない

ウ　A は C を含むとも含まないとも決められない

[(c)に関する解答群]

ア　A と B の論理和は偽である　　　　イ　A と B の論理積は真である

ウ　A の否定は真である　　　　　　　エ　A と B の排他的論理和は真である

4.4 ビット演算に関する次の記述中の( )に入れるべき適当な字句を、解答群の中から選べ。

8 ビット(16 進数で表示すると 2 けた)からなるコード X がアキュムレータにあるとする。次の(1)～(5)は、アキュムレータ上の X ビットごとの論理演算をほどこして、他のコードを生成する処理である。

(1)X の最上位ビット(1 番左側のビット)を 0 とするには、(a)。他の 7 ビットは変化させないものとする。

(2)X の最上位ビットを 1 とするには、(b)、他の 7 ビットは変化させないものとする。

(3)X の下位 4 ビットを変化させず、上位 4 ビットをすべて 0 とするには、(c)。

(4)X の上位 4 ビットを変化させず、下位 4 ビットをすべて 1 とするには(d)。

(5)X の全てのビットを反転させるには、(e)。

a=ウ b=カ c=ア d=イ e=ケ

解説：

まずは、２進数になおしてみる。

0F=00001111
7F=01111111
80=10000000
FF=11111111

そして、例として、『00101010』を計算する。

ア=00001010	上位４ビットが０に変化。下位４ビットが変化なし。
イ=10101111	上位４ビットは変化なし。下位４ビットは１に変化。
ウ=00101010	変化なし
エ=01111111	最上位ビット以外のすべてのビットが１に変化。
オ=00000000	すべてのビットが０に変化。
カ=10101010	最上位ビットが１に変化。
キ=00101010	変化なし。
ク=11111111	すべてのビットが１に変化。
ケ=11010101	

この実験より、a=ウまたはキ、b=カ、c=ア、d=イ、e=ケとなる。a は、違う数字で実験すると、ウということが分かる。

[解答群]

ア　16 進数 0F と AND をとる

イ　16 進数 0F と OR をとる

ウ　16 進数７F と AND をとる

エ　16 進数７F と OR をとる

オ　16 進数 80 と AND をとる

カ　16 進数 80 と OR をとる

キ　16 進数 FF と AND をとる

ク　16 進数 FF と OR をとる

ケ　16 進数 FF と EOR(Exclusive Or)をとる