警告:このファイルを、俺の意に反して広めないこと。俺の意とは、アンケートに協力してくださった方だけに感謝の気持ちでファイルを公開したいから、それ以外の人に見せるのはしばらくしてからにしたい、というものである。

また、内容については責任を取らない(特に左辺値と右辺値)。

# 1.講義スライドの「C言語のポイント」を参考にして作成した問題

問題1

穴埋めせよ。①と②を答えてしまえば、後は思ったより少ない。

左辺値は＿①＿と考えてよい。一方、右辺値は＿②＿である。

(左辺値)=(右辺値);

は、

「(左辺値)」という名の＿①＿に、「(右辺値)」と書かれた＿②＿を入れるようなものである。

x=5;

は

＿①＿xに＿②＿5を入れる操作である。

x=5;

y=x;

は、

＿①＿xに＿②＿5をいれ、それから＿①＿yに＿①＿x**の中に入ってる＿②＿5**を入れる操作である。

今太字で示したが、＿①＿yに＿①＿xを入れる訳でない。

これは、「評価」を「紙を見たり、計算したり、＿③＿してくること」と考え、代入を「＿②＿を＿①＿に入れること」と考えればよいだろう[1]。

したがって、

(左辺値)=(左辺値);

という式もありえる。

しかし、

5=x;

のようなものは、＿②＿5に＿①＿xの中の＿②＿を入れろという無茶になるため、あり得ない。つまり左辺値(＿①＿)は、＿①＿から＿②＿を引っ張り出すことで右辺値(＿②＿)になることができるが、右辺値(＿②＿)は左辺値(＿①＿)にはなれないのである。

x=y=6は、=が＿④＿の演算子といって、＿⑤＿側を優先したがる性質があるので、

x=(y=6)のように扱われる。

＿⑥＿る操作であると考えてよいだろう。

C言語では、評価の結果は右辺値とすることになっている。つまり、＿①＿に＿②＿を入れた結果は、＿⑦＿であるということだ。よって、＿①＿xにも＿②＿6が入る。しかし、(x=y)=6は、＿⑧＿ることになり、

＿②＿に＿②＿6を入れるという無理な要求になってしまう。よって不適切である。

\*ptrというのは、ptrという＿⑩＿るものである。つまり、普通の＿①＿と見なせる。よって、

\*ptr=7というのは可能である。

問題2

++i++が何故ダメなのか、「副作用」を説明しつつ、評価、副作用、定まらないから という言葉を用いて言え。

問題3

副作用を起こす演算子の例を挙げよ。

問題4

10<9<8の真偽を説明せよ。

問題5

for(①;②;④)

{

③

}

をwhile文で書き換えよ。また、while文に書き換えられない条件を説明せよ。

問題6

while(①)

{

②

}

をfor文で書き換えよ。

問題7

配列は0-オリジンである。どういう意味か。

問題8

要素n個の配列の各要素について扱うfor文を、昇順のものと降順のもの1つずつかけ。

答え1

① 箱,② 紙,③ 箱から紙を引っ張り出,④ 右結合,⑤ 右,⑥ 箱xに、「箱yに紙6を入れたときの結果」を入れ,

⑦ 箱,⑧ 「箱xに箱ｙの中の紙を入れた結果」に紙6を入れ,⑨ 箱,⑩ ポインタを変数に変え

答え2

評価によって値が変わることを副作用という。++iやi++だけでは副作用が発生せず、++i;やi++;として初めて発生する。従って、++i++では副作用が発生する前に別の副作用がまた発生しようとしてしまうため、副作用が最終的にどうなるかが定まらないから

答え3

+=,-=,/=,\*=,?=,++,--

答え4

偽は0、真は非0の代表値として1であらわされるので、

10<9<8

偽<8

0<8

真

答え5

①

while(②)

{

③

④

}

但し、③が「continue;」だった場合、for文では④が実行され、while文では実行されないので、同じことにはならない。

答え6

for(;①;)

{

②

}

答え7

0から数え始める。

答え8

for(i=0;i<n;i++)

{

}

と

for(i=n-1;i>=0;i--)

{

}

[1] <https://www.ibm.com/support/knowledgecenter/ja/ssw_ibm_i_72/rzarg/lvalue.htm> によれば、

*すべての代入演算子は、それらの右方オペランドを評価し、その値を左方オペランドに代入します*

とのこと。

# 2.各講義スライドを参考に作成した問題

## 第一回講義

問題9

gccで、raw.cをapp.exeという名前でコンパイルしたいとき、どう打てばよいか。また、a.exeという名前でコンパイルしたいときはどうするか。

問題10

if文、for文などで出てくる{　～　}は何か。

問題11

{　～　}を用いないif文はどんな振る舞いをするか。

問題12

‘¥x41’がA,’\x61’がaの時、文字コードのみを用いて「Clang」と表示させるprintfをかけ。

問題13

穴埋めせよ。

int x=10;

printf("10進数＿①＿を８進で表すと＿②＿, 16進で表すと%＿③＿であ る。¥n",x,x,x);

問題14

問題13で、10進数を8進数、16進数に変換したわけだが、8進数や16進数の10を変換したいときには、  
int x=何 にすればよいか。

問題15

変数の初期化とは何か

問題16

問題1でも取り上げたとおり、変数=変数とすると、左辺にある変数はそのまま扱われるのに対して、

右辺にある変数は、その中にある右辺値を取り出すのに使われる。  
このように、場所によって変数に対する振る舞いが異なるが、これと関連してc言語以降の代入にまつわる不可解な点を「振る舞いが異なる」という言葉を用いて説明せよ。

答え9

gcc -o app raw.c

但し、a.exeという名前でよいときは

gcc raw.c

でよい。

答え10

ブロック。**ブロックも文の一種**だが、例外的に最後に「;」がつかない。複数の文をまとめるのに使う。

答え11

if()の直後の文一つだけが、if内の条件式が正しいときに実行される。他はすべて必ず実行される。

答え12

printf(“%c%c%c%c%c”, ‘\x43’, **’\x6c’**, ’\x61’, ‘\x64’, ’\x67’);

\x42→B,\x43→C…となることを類推したい。但し、16進数であることに気をつけよ(太字部)。

答え13

① %d,② %o,③ %x

それぞれDecimal, Octal, heXadecimalの意味だと思われる。

答え14

8進数→010

16進数→0x10

答え15

int i=5;

など、変数宣言と同時に値を決めること。**代入とは異なる**。

答え16

右辺が変数の場合と右辺が定数の場合で、コンパイラの振る舞いが異なる。