



プログラミング言語 第三回レポート

担当: 篠沢 佳久
栗原 聡



第三回レポート課題

問題①②③④

問題はそれぞれどれか一問を解いて下さい

report-3.zipをダウンロード, 解凍し, 3個のファイルをプログラムと同じフォルダーに置いて下さい



第三回レポート課題

- 課題：問題①②③④
 - 問題①②③は(a) (b)のどちらかを解いて下さい
 - 問題④は(a) (b) (c)のどれか一問を解いて下さい
 - 難易度は(a) < (b) < (c)ですので配点も(a) < (b) < (c)とします
- 締め切り
 - 7/22(月) 10時半(講義開始前まで)



問題①-(a)

問題①は(a)もしくは(b)のどちらかを回答して下さい

- 2019年の暦において, 月日を入力し, 入力した月日が1月1日から数えて何日目であるかを印字するプログラムを書きなさい.
 - 入力する月日は暦上, 正しい日付を入力するものとします
 - 2月は28日です
 - 1月1日は1日目, 12月31日は365日目です

```
> python report-3-a.py
月を入力して下さい > 7
日を入力して下さい > 11
1月1日から7月11日までは192日です
```

```
> python report-3-a.py
月を入力して下さい > 12
日を入力して下さい > 31
1月1日から12月31日までは365日です
```



問題①-(b)

問題①は(a)もしくは(b)のどちらかを回答して下さい

- 2019年の暦において、月日を入力し、入力した月日が5月1日（令和初日）から何日後もしくは5月1日の何日前かを印字するプログラムを書きなさい。
 - 入力する月日は暦上、正しい日付を入力するものとします
 - 2月は28日です
 - 5月2日は1日後、4月30日は1日前です



問題①-(b) (表示例)

```
> python report-3-b.py  
月を入力して下さい > 4  
日を入力して下さい > 23  
4月23日は5月1日の8日前です
```

```
> python report-3-b.py  
月を入力して下さい > 12  
日を入力して下さい > 31  
12月31日は5月1日から244日後です
```

```
> python report-3-b.py  
月を入力して下さい > 5  
日を入力して下さい > 1  
5月1日は令和初日です
```

問題②-(a)

問題②は(a)もしくは(b)のどちらかを回答して下さい

- アルファベット26文字(小文字)の文字列を下記のような変換表を用いて変換(暗号化)することを行ないます($k=8$ のシーザー暗号)

平文	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
暗号文	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h

- 例えば, 文字列「program」はこの変換表によって, 「xzwoziu」と変換されます
- 1年生の時に習いましたね? 平文から暗号文に変換することを暗号化, 暗号文から平文に戻す変換を復号化と呼びます.



問題②-(a)

問題②は(a)もしくは(b)のどちらかを回答して下さい

- 2文字以上の文字列cをキーボードから入力し、変換表に従って文字列を変換した後、印字するプログラムを作成しなさい.

```
> python report-3-2-a.py
平文を入力して下さい > program
平文  -> program
暗号文 -> xzwoziu
```

```
> python report-3-2-a.py
平文を入力して下さい > administration
平文  -> administration
暗号文 -> iluqvqabzibqwv
```




問題②-(b)

問題②は(a)もしくは(b)のどちらかを回答して下さい

- 問題②-(a)において, k (整数値, $0 < k < 26$)および2文字以上の文字列 c をキーボードから入力し, k の値によって, 文字列 c を変換(暗号化)するプログラムを作成しなさい.
- さらに暗号文を復号化し, 元の文字列に戻ることを確認しなさい.
- 変換表は表示しなくてもかまいません.

問題②-(b) (表示例)

```

C:\Windows\system32\cmd.exe
C:\home\shino\Program-2019\report>python report-3-2-b.py
k > 3
[ 変換表 ]
-----
|abcdefghijklmnopqrstuvwxyz|
|defghijklmnopqrstuvwxyzabc|
-----

平文を入力して下さい > program
暗号化前 -> program
暗号化後 -> surjudp
復号化後 -> program
  
```

kを入力

k=3の変換表

単語を入力

k=3の変換表

変換前	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
変換後	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c

問題②-(b) (表示例)

```

C:\Windows\system32\cmd.exe
C:\home\shino\Program-2019\report3\python report-3-2-b.py
k > 10
[ 変換表 ]
-----
|abcdefghijklmnopqrstuvwxyz|
|klmnopqrstuvwxyzabcdefg|
-----
平文を入力して下さい > program
暗号化前 -> program
暗号化後 -> zbyqbkw
復号化後 -> program
  
```

kを入力

k=10の変換表

単語を入力

k=10の変換表

変換前	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
変換後	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f	g	h	i	j



問題③-(a)

問題③は(a)もしくは(b)のどちらかを回答して下さい

- file1.txtには, レポート課題を提出した学生の学籍番号*が記載されています.
- 学籍番号を入力し, レポートが提出されているか, されていないかを調べるプログラムを作成しなさい.
- プログラムとfile1.txtは同じフォルダーに置いて下さい.

*乱数で作りました



問題③-(a) (表示例)

提出表にある場合

```
> python report-3-3-a.py  
学籍番号? > 61919927  
学籍番号61919927くんは提出済です
```

提出表にない場合

```
> python report-3-3-a.py  
学籍番号? > 61918900  
学籍番号61918900くんは未提出です
```



問題③-(b)

問題③は(a)もしくは(b)のどちらかを回答して下さい

- file1.txtには, プログラミング言語の履修者の学籍番号*が記載されています.
- file2.txtには, レポート課題の提出者の一覧*が記載されています.
- レポート課題をまだ提出していない学生の学籍番号を全員印字するプログラムを作成しなさい.

*乱数で作りました



問題③-(b)

問題③は(a)もしくは(b)のどちらかを回答して下さい

- file1.txtとfile2.txtはpythonプログラムと同じフォルダーに置いて, 読み込んで下さい.



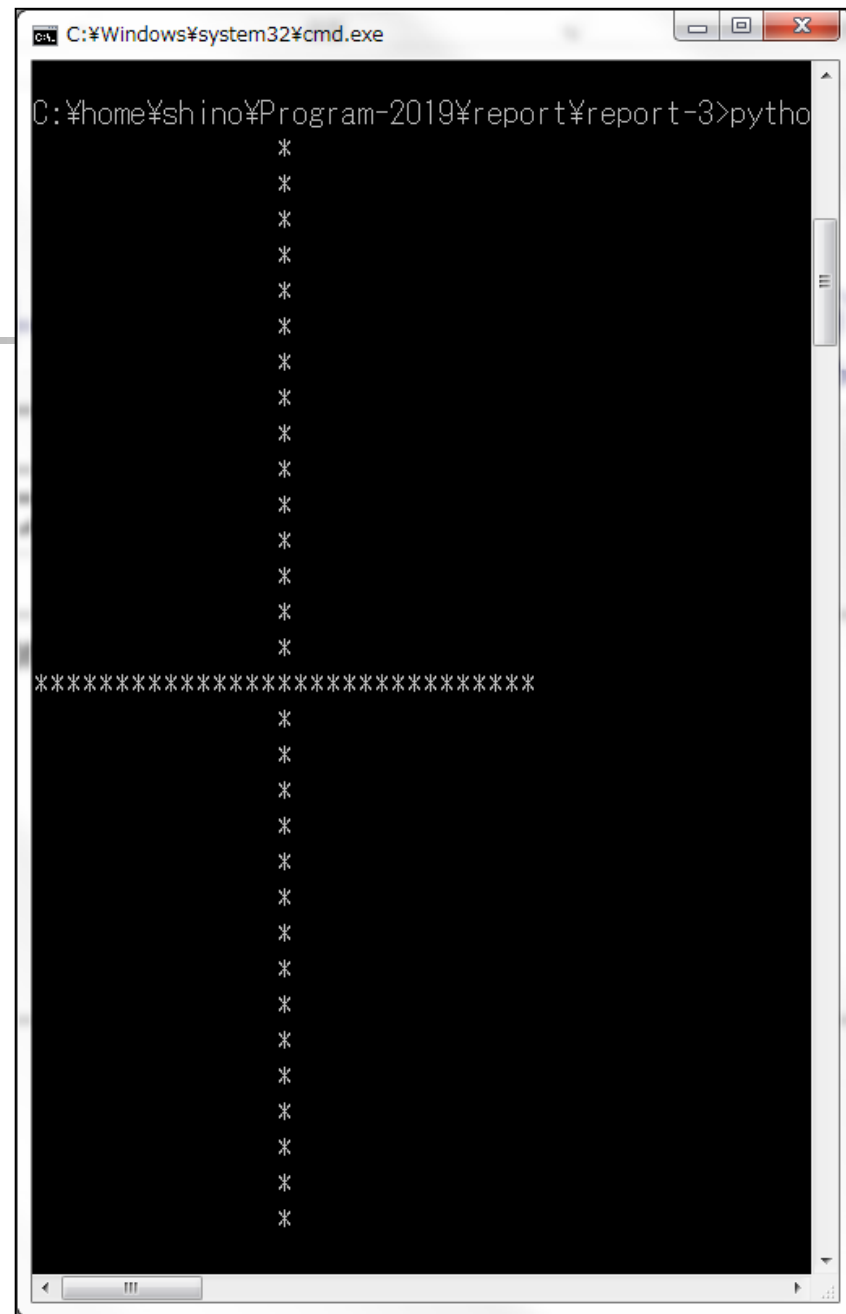
問題③-(b) (実行例)

```
> python report-3-3-b.py  
未提出の学生は...
```



問題④

- 次ページのサンプルプログラムを実行すると、右図のような結果となります



```
C:\Windows\system32\cmd.exe
C:\home\shino\Program-2019\report\report-3>python
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*****
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
```



問題④のサンプルプログラム①

```
a=[0]*31
for i in range(31):
    a[i]=[1]*31

for y in range(len(a)):
    for x in range(len(a[y])):
        if x == 15 or y == 15:
            a[y][x]=1
        else:
            a[y][x]=0
```



サンプルプログラム②

サンプルプログラム①の続き

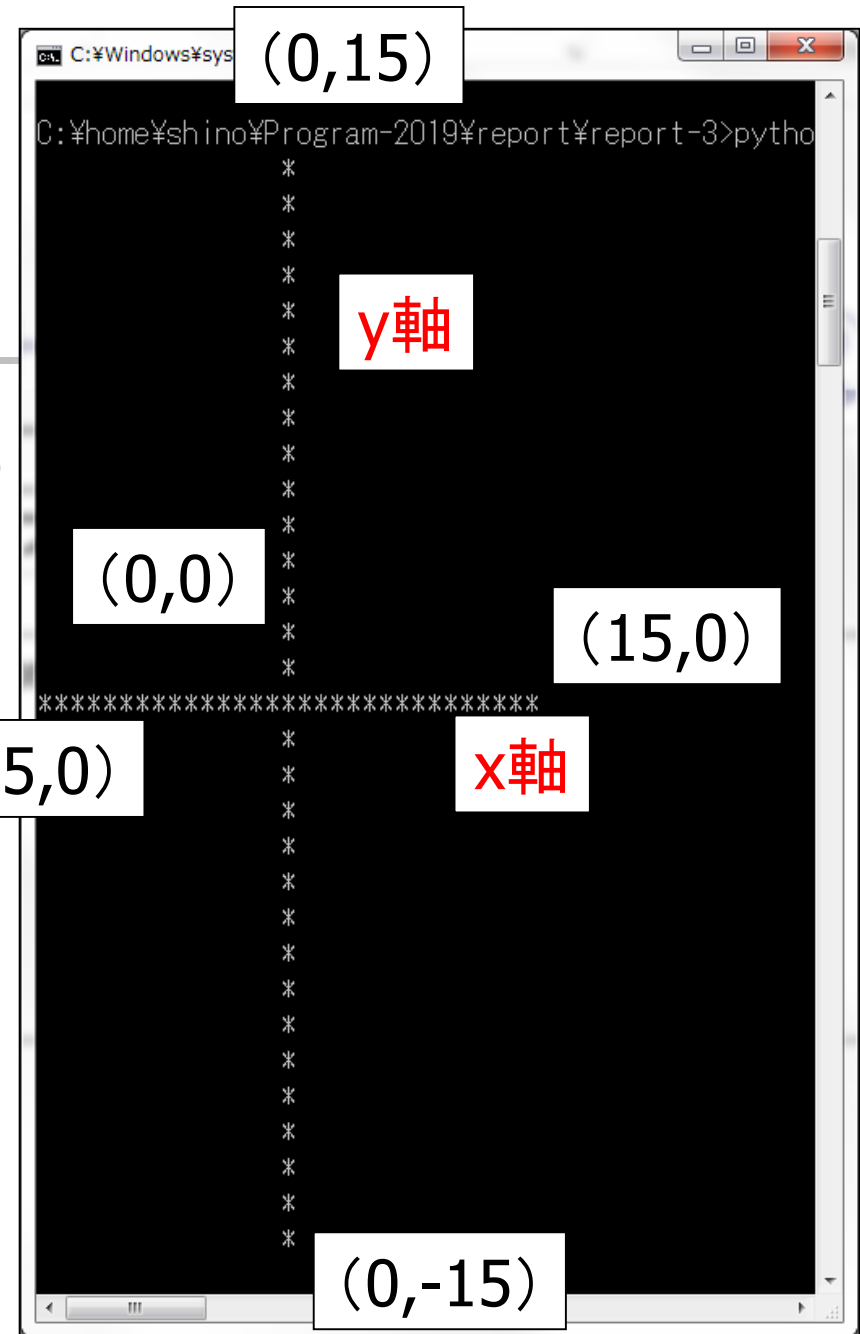
```
for y in range(len(a)):
    for x in range(len(a[y])):
        if a[y][x]==1:
            print( "*" , end="" )
        else:
            print( " " , end="" )
    print()
```

問題④

- 実行結果は、右図のように二次元座標を表わすものとします

- $-15 \leq x \leq 15$

- $-15 \leq y \leq 15$





問題④-(a)

問題④は(a) (b) (c)のどれか一問を回答して下さい

- $x=-15, -14, \dots, 14, 15$ において, 座標軸上に $y=x$ および $y=-x$ となる点をプロットしなさい
- 次ページのようにそれぞれ別々に表示させるプログラムを作成しなさい
- 点は "*" 以外で印字して下さい



問題④-(b)

問題④は(a) (b) (c)のどれか一問を回答して下さい

- $x = -15, -14, \dots, 14, 15$ において, 座標軸上に $y = nx + m$ となる点をプロットしなさい
- n, m は整数として, キーボードから読み込めるようにしなさい
- y の範囲は $-15 \leq y \leq 15$ としているので, 範囲外は印字しないようにすること
- 点は "*" 以外で印字して下さい



問題④-(C)

問題④は(a)(b)(c)のどれか一問を回答して下さい

- $x = -15, -14, \dots, 14, 15$ において, 座標軸上に $y = nx + m$ となる点をプロットしなさい
- さらに $y = nx + m$ と $y = 0$ で囲まれる領域もプロットしなさい
- n, m は整数として, キーボードから読み込めるようにしなさい
- y の範囲は $-15 \leq y \leq 15$ としているので, 範囲外は印字しないようにすること
- 点は "*" 以外で印字して下さい(次頁参考)

C:\Windows\system32\cmd.exe

C:\home\shino\Program-2019\report\report-3>python

```
n > 2
m > 3
[ y=2x+3のグラフ ]
*      P
*      C
*      PC
*      CC
*      PCC
*      CCC
*      PCCC
*      CCCC
*      PCCCC
*      CCCCC
*PCCCCC
*CCCCCC
PCCCCC
CCCCC
PCCCCC
*****
CCCCCCP *
CCCCCC  *
CCCCCP  *
CCCCC   *
CCCCCP  *
CCCCC   *
CCCCP   *
CCCC    *
CCCP    *
CCC     *
CCP     *
CC      *
CP      *
C       *
P       *
```

$y=2x+3$

C:\home\shino\Program-2019\report\report-3>

C:\¥Windows¥system32¥cmd.exe

C:\¥home¥shino¥Program-2019¥report¥report-3>python

```
n > -3
m > -2
[ y=-3x+-2のグラフ ]

      *
      *
P     *
C     *
C     *
CP    *
CC    *
CC    *
CCP   *
CCC   *
CCC   *
CCCP  *
CCCC  *
CCCC  *
CCCP*
```

$y = -3x - 2$

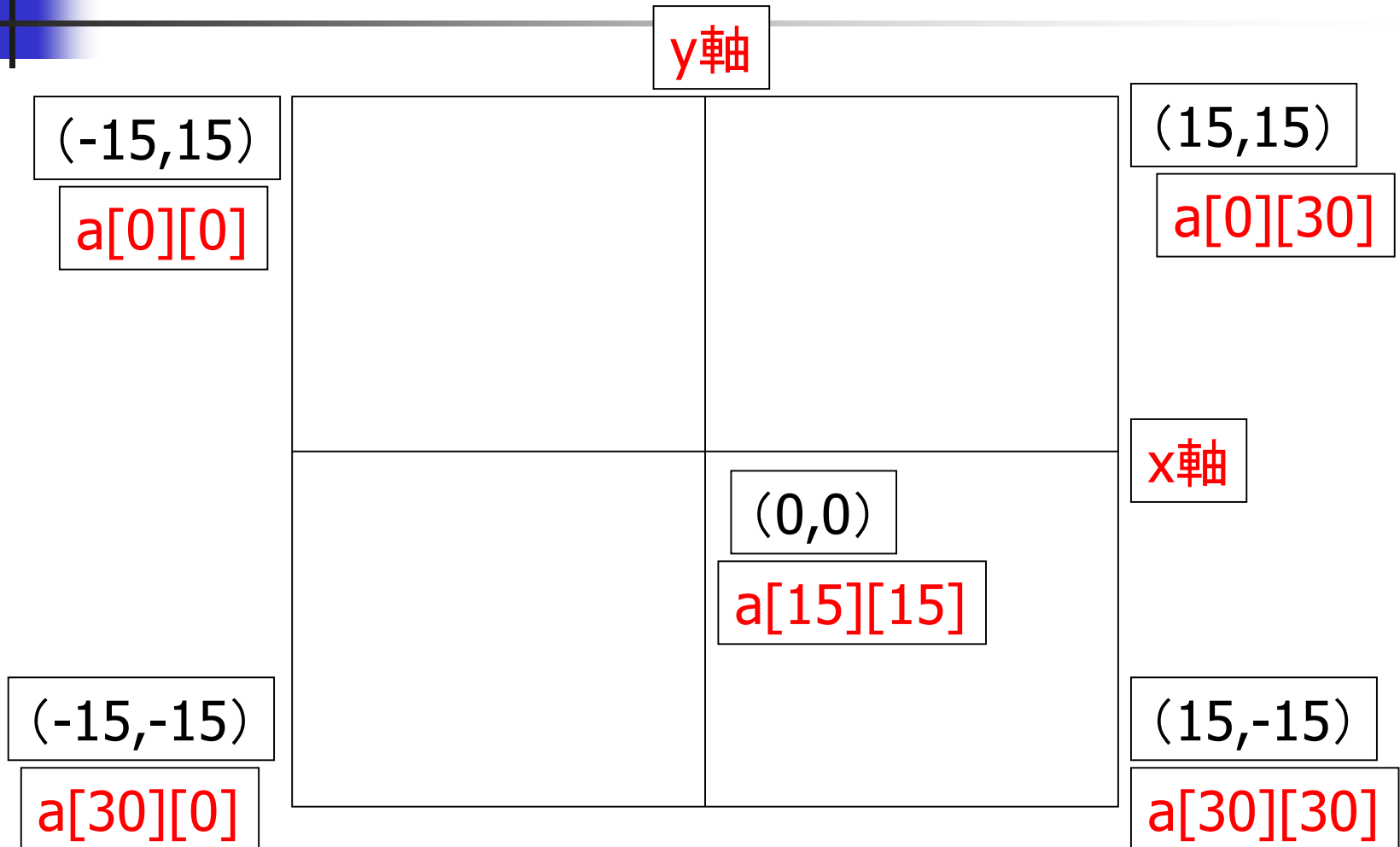
```
CCCCC
PCCCC
*CCCC
*CCCC
*PCCC
* CCC
* CCC
* PCC
* CC
* CC
* PC
* C
* C
* P
*
```

別々の記号でプロットして下さい

C:\¥home¥shino¥Program-2019¥report¥report-3>

26

ヒント(座標軸と配列の関係は?)





提出方法

- 「keio.jp」の「教育支援システム」で提出
 - レポート(ワープロ), プログラムファイルおよび実行結果を「第三回レポート課題」に電子的に提出して下さい
 - レポート本文
 - 先頭の一行は, 「プログラミング言語第三回レポート」とし, 以下に学籍番号・氏名を記して下さい
 - それ以下に, 課題内容, 各プログラムの説明とプログラムをおよび実行結果, そしてプログラムを書く上で工夫した点, 悩んだ点(考察)を書いて下さい
 - 最後に, 感想および意見を書いて下さい



レポートの書き方①

A4

プログラミング言語 第三回レポート
学籍番号 氏名

1. 課題内容

何を目的としたプログラムか
入力に対してどのような出力をするプログラムなのか

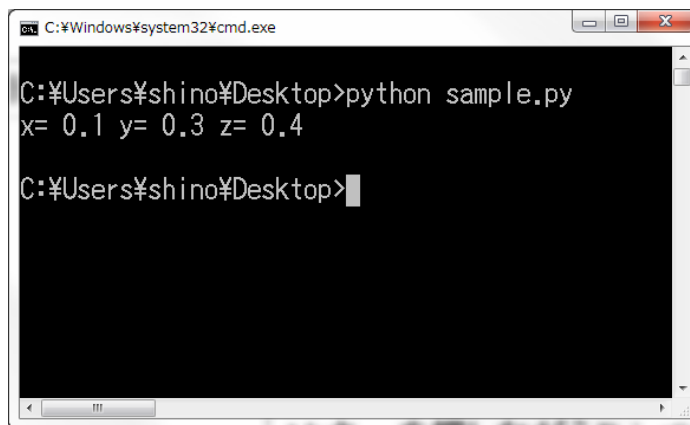
2. プログラムの説明

使用した変数の説明
問題を解く上でどのようにプログラミングしたのか
プログラムの各行の説明
(ただしプログラムが長い場合、一行ごとに説明する
必要はない、まとまった部分ごとで説明すればよい)

レポートの書き方②

3. プログラムと実行結果

```
x = 0.1
n = 1
y = 0.3
if n==1: z=x+y
if n!=1: z=x*y
print( "x=", x , "y=", y , "z=", z )
```



A screenshot of a Windows command prompt window. The title bar reads "C:\Windows\system32\cmd.exe". The command prompt shows the following text: "C:\Users\shino\Desktop>python sample.py", followed by the output "x= 0.1 y= 0.3 z= 0.4", and then the prompt "C:\Users\shino\Desktop>" with a cursor.

```
C:\Windows\system32\cmd.exe
C:\Users\shino\Desktop>python sample.py
x= 0.1 y= 0.3 z= 0.4
C:\Users\shino\Desktop>
```



レポートの書き方③

4. 考察

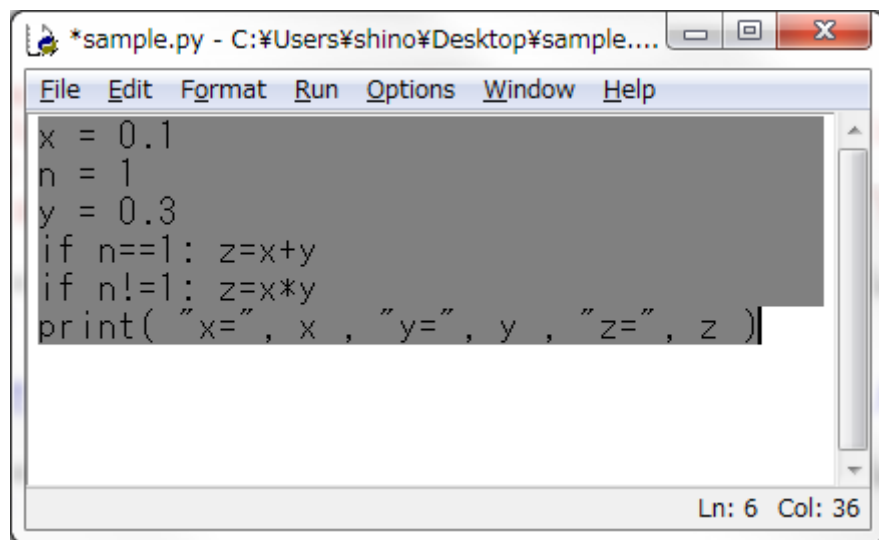
目的とした出力が適切に得られているか
プログラムの作成上において工夫した点
プログラムの作成上において悩んだ点(分からなかった点)
自分がこの課題を作成する上で苦労した点,
他の人とは違うという点を主張して下さい

5. 感想

意見, 要望など

プログラムと実行結果をMS-Word への貼り付け方①

① エディター上にてプログラムを選択



The screenshot shows a window titled '*sample.py - C:\Users\shino\Desktop\sample....'. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code in the editor is:

```
x = 0.1
n = 1
y = 0.3
if n==1: z=x+y
if n!=1: z=x*y
print( "x=", x , "y=", y , "z=", z )
```

The status bar at the bottom indicates 'Ln: 6 Col: 36'. A right-click context menu is open over the code, with 'コピー(C)' (Copy) highlighted.

② 右クリック→「コピー」

元に戻す(U)

切り取り(T)

コピー(C)

貼り付け(P)

削除(D)

すべて選択(A)

右から左に読む(R)

Unicode 制御文字の表示(S)

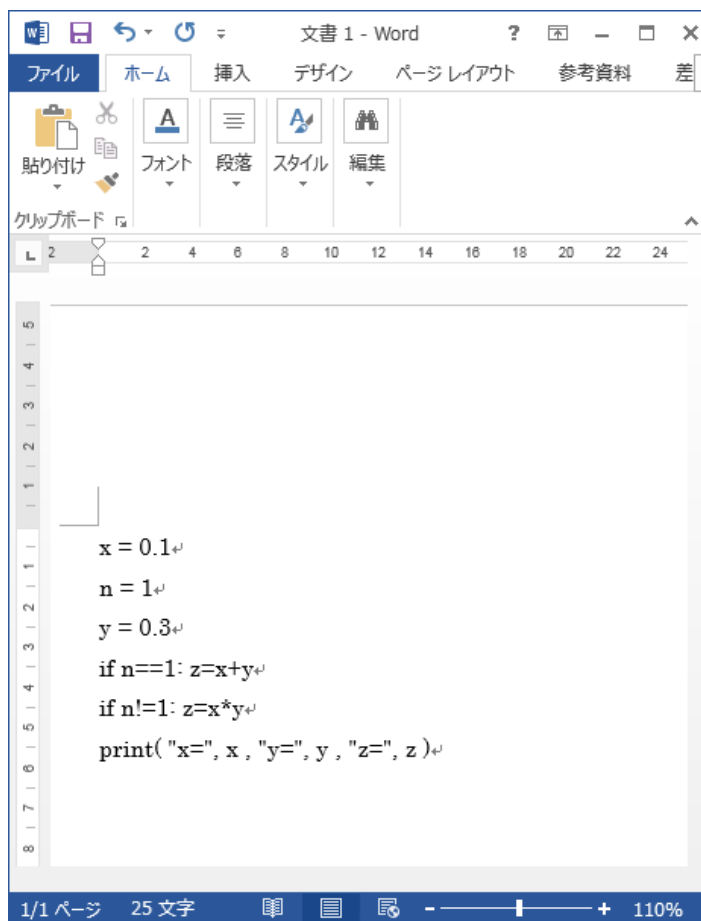
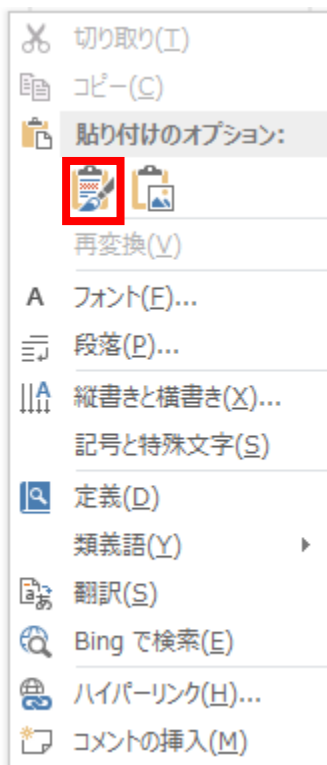
Unicode 制御文字の挿入(I) ▶

IME を開く(O)

再変換(B)

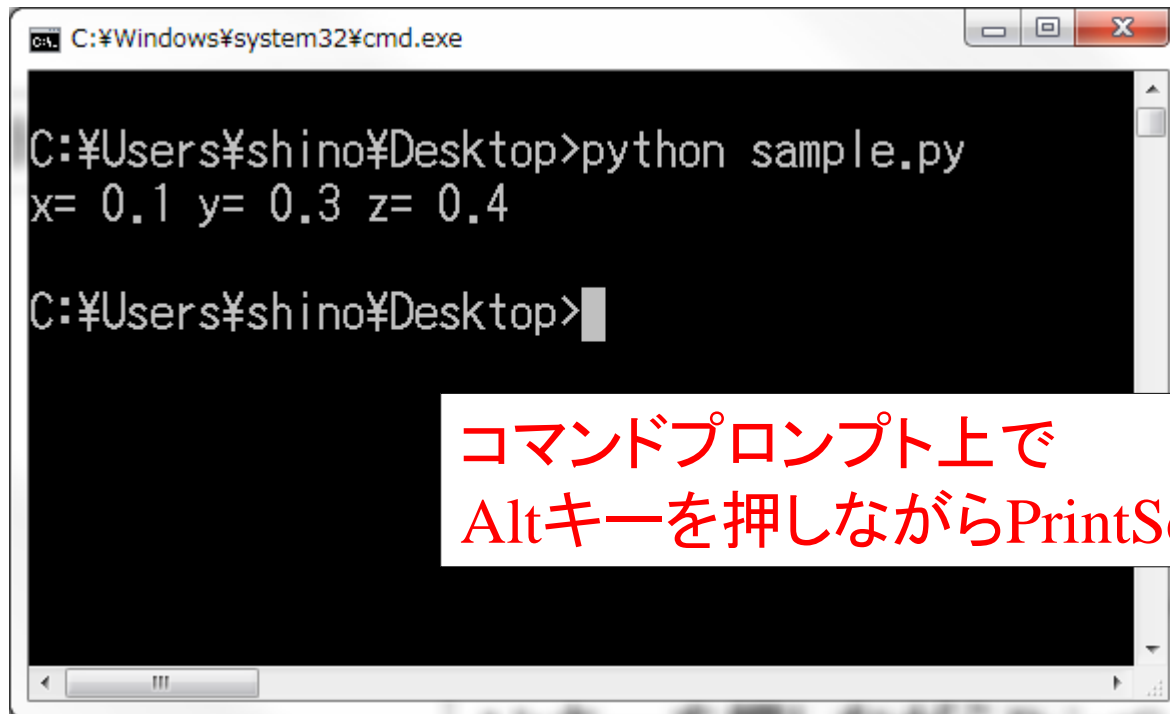
プログラムと実行結果をMS-Wordへの貼り付け方②

③ MS-Word上で右クリック
→「貼り付け」



プログラムと実行結果をMS-Word への貼り付け方③

実行結果



The screenshot shows a Windows Command Prompt window titled "C:\Windows\system32\cmd.exe". The prompt is at "C:\Users\shino\Desktop>". The user has entered the command "python sample.py", and the output is "x= 0.1 y= 0.3 z= 0.4". The prompt is now at "C:\Users\shino\Desktop>".

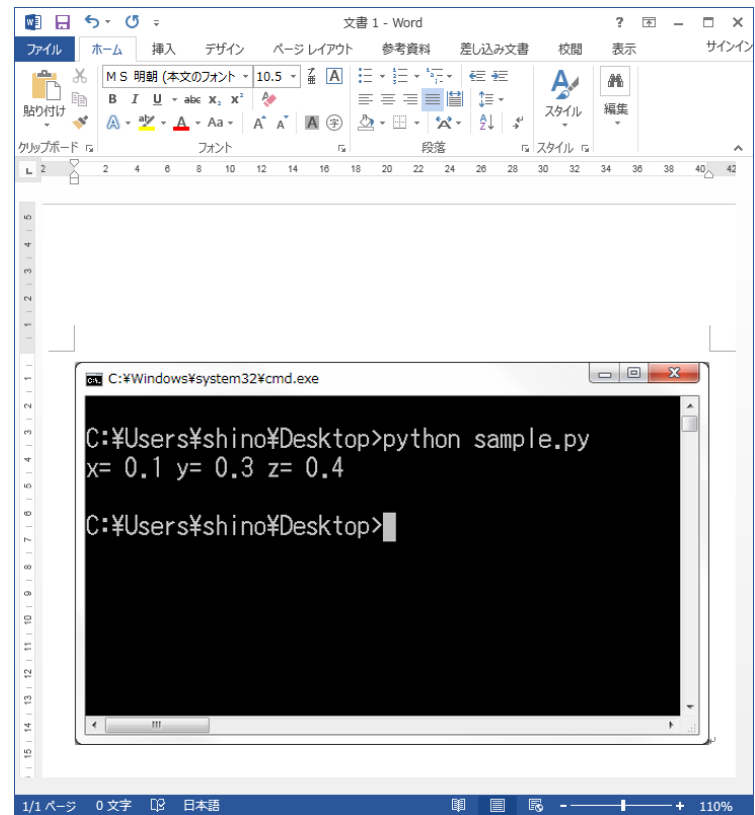
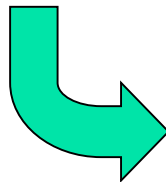
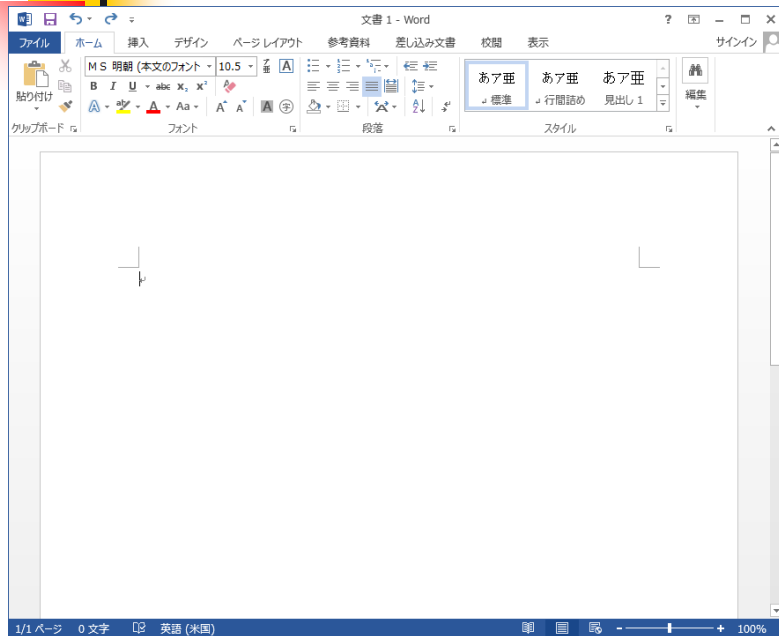
```
C:\Windows\system32\cmd.exe

C:\Users\shino\Desktop>python sample.py
x= 0.1 y= 0.3 z= 0.4

C:\Users\shino\Desktop>
```

コマンドプロンプト上で
Altキーを押しながらPrintScrn

MS-Word上で右クリック
→「貼り付け」



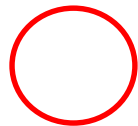
コマンドプロンプトの画面が貼り付けられる



レポートを書く上での注意①

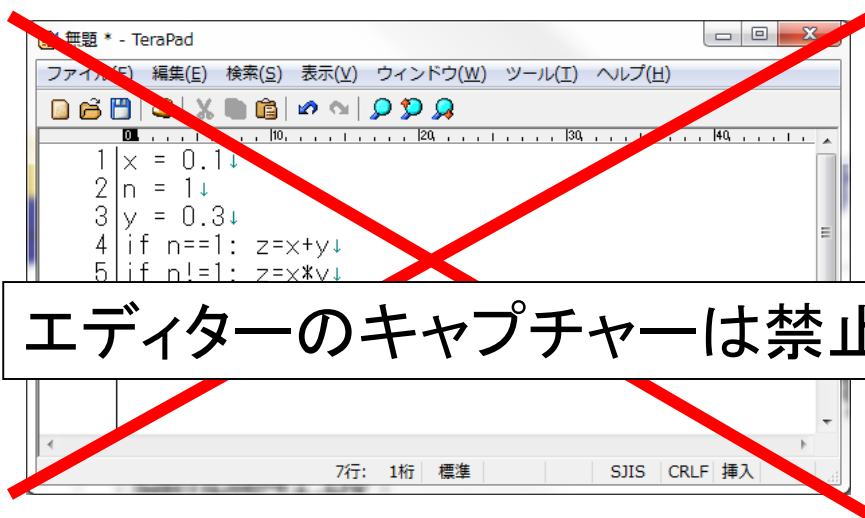
- プログラムはレポート中に「テキスト」にて貼り付けて下さい
- エディターのキャプチャー画面を貼りつけないで下さい(こちらは減点します)
- 採点の際に、皆さんのプログラムを実行するためです

レポートを書く上での注意②



```
x = 0.1  
n = 1  
y = 0.3  
if n==1: z=x+y  
if n!=1: z=x*y  
print( "x=", x , "y=", y , "z=", z )
```

プログラムは「テキスト」
にて貼り付けて下さい



エディターのキャプチャーは禁止



レポートを書く上での注意③

- プログラム, レポートのコピーは厳禁
- 相談するのはかまわない
- 完成しなかった場合, どこまでできたのか, うまく動かなかったこと, 悩んだことを考察に記述して下さい