クラス	学籍番号	氏名
Cプログラミング 入門(電生 2)		

次のコードの実行結果(出力)を書け。

```
#include <stdio.h>

int main(void)
{
    printf("%04d\n", 0);
    printf("%02d%02d\n", 2025, 7);
    printf("%.0f\n", 2025.7);
}
```

```
#include <stdio.h>
#define SQUARE(x) ((x) * (x))

int main(void)
{
    printf("%d\n", SQUARE(5));
    printf("%.2f\n", SQUARE(0.1));
    printf("%d\n", SQUARE(1 + 4));
}
```

```
#include <stdio.h>
#define COUNTOF(a) ((int)(sizeof(a) / sizeof((a)[0])))

int main(void)
{
   int vals[6] = { 5, 4 };
   for (int i = 0; i < COUNTOF(vals); ++i)
      {
       printf("%d\n", vals[COUNTOF(vals) - 1 - i]);
   }
}</pre>
```

```
#include <stdio.h>
#include <stdbool.h>
bool A()
{
    puts("A");
    return true;
}
bool B()
    puts("B");
    return false;
}
int main(void)
{
    if (A() || B())
        puts("C");
    }
    if (A() && B() && A())
        puts("D");
    }
}
```

```
puts("C");
}

if (A() && B() && A())
{
   puts("D");
}

A
C
A
B
```

```
#include <stdio.h>
int main(void)
{
    int n = 15;
    n += 5;
    printf("%d\n", n);
    n *= n;
    printf("%d\n", n);
    ++n;
    printf("%d\n", n);
}

20
400
401
```

```
#include <stdio.h>

typedef enum
{
    RED = 100,
    GREEN = 102,
    BLUE,
    YELLOW
} Color;

int main(void)
{
    printf("%d\n", BLUE);
    printf("%d\n", YELLOW);
}

103
```

# (参考)ASCII 表

	(参考) ASCII 农										
10進	16進	文字	10進	16進	文字	10進	16進	文字	10進	16進	文字
0	00	∖0 (ヌル文字)	32	20	space	64	40	@	96	60	,
1	01	SOH	33	21	!	65	41	Α	97	61	a
2	02	STX	34	22	=	66	42	В	98	62	b
3	03	ETX	35	23	#	67	43	С	99	63	U
4	94	EOT	36	24	\$	68	44	D	100	64	d
5	05	ENQ	37	25	%	69	45	E	101	65	е
6	96	ACK	38	26	&	70	46	F	102	66	f
7	07	\a (ベル)	39	27	-	71	47	G	103	67	g
8	98	<b>\b</b> (バックスペース)	40	28	(	72	48	Н	104	68	h
9	09	<b>\t</b> (水平タブ)	41	29	)	73	49	I	105	69	i
10	0A	<b>\n</b> (改行)	42	2A	*	74	4A	J	106	6A	j
11	0B	∖v(垂直タブ)	43	2B	+	75	4B	К	107	6B	k
12	9C	<b>\f</b> (改ページ)	44	2C	,	76	4C	L	108	6C	ι
13	0D	\r (復帰)	45	2D	ı	77	4D	М	109	6D	m
14	0E	S0	46	2E		78	4E	N	110	6E	n
15	0F	SI	47	2F	/	79	4F	0	111	6F	0
16	10	DLE	48	30	0	80	50	Р	112	70	р
17	11	DC1	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	DC3	51	33	3	83	53	S	115	73	5
20	14	DC4	52	34	4	84	54	Т	116	74	t
21	15	NAK	53	35	5	85	55	U	117	75	u
22	16	SYN	54	36	6	86	56	V	118	76	<b>v</b>
23	17	ETB	55	37	7	87	57	W	119	77	W
24	18	CAN	56	38	8	88	58	X	120	78	х
25	19	EM	57	39	9	89	59	Υ	121	79	у
26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	;	91	5B	[	123	7B	{
28	10	FS	60	3C	<	92	5C	\	124	7C	
29	1D	GS	61	3D	=	93	5D	]	125	7D	}
30	1E	RS	62	3E	>	94	5E	^	126	7E	2
31	1F	US	63	3F	?	95	5F	_	127	<b>7</b> F	DEL

次のコードを実行し、指定した入力(複数行)を一度に与えた場合の出力を書け。

```
#include <stdio.h>
                                                    入力
#include <ctype.h>
                                               Tokyo 2020
                                               Paris 2024
int main(void)
                                               LA 2028
{
    int n = 0;
                                               n = 4
    for (;;)
        char c = (char)getchar();
        if (c == '\n')
            break;
        n += (isdigit(c) ? (c - '0') : 0);
    printf("n = %d\n", n);
}
```

次の式の結果について、何型でどのような値になるかを書け。

	型(int など)	值
5 / 2	int	2
pow(5.0, 0.5) <= 5.0	int	1
2 ? (3 ? 4 : 5) : 6	int	4
fabs(81.0)	double	81.0
strlen("\\\\n")	size_t	3
isxdigit('x') && true	int	0

## 次の説明に合う関数を最下段の選択肢からすべて選んで書け。

渡した文字が小文字であるかを返す	isdigit
渡した double 型の値の絶対値を返す	fabs
整数値を渡すと、それに対応する ASCII 文字を出力する	putchar, printf
ヌル終端の前までにある char 型の要素数を返す	strlen
乱数生成器のシード値を、指定した整数に変更する	srand

複数の引数をとる printf, scanf, pow, hypot

<stdio.h> の関数 puts, printf, scanf, getchar, putchar

戻り値が double 型 fabs, ceil, floor, pow, hypot

puts printf scanf getchar putchar isdigit islower isalnum rand srand abs fabs ceil floor pow hypot strlen

誤っている部分があれば下線を引き、それを置き換える正しい記述を書け。 誤っている部分が無い場合は「正しい」と書け。

変数nが0以上10以下であるかは、0くnく10という式で判定する

 $0 \le n \& n \le 10$ 

配列 int n[10]; の先頭要素は n[1] で、末尾の要素は n[10]

n[0] で、末尾の要素は n[9]

1つの構造体が持つメンバ変数は、すべて同じ型にする必要がある

それぞれ異なる型でもよい

scanfで使う double 型用の書式指定子は %1f (パーセントいちエフ)

%1f (パーセントエルエフ)

プロトタイプ宣言では引数名を書かなくてもよい

正しい

# 要約して本質的な機能を説明せよ。 | 行ずつ説明した場合は不正解とする。

```
与えられた文字列に「数
bool Check(const char s[])
                                    字」「小文字」「大文
   int digit = 0, lower = 0, upper = 0; |字」がそれぞれ | つ以上
                                    含まれているかを判定し
   for (int i = 0; s[i] != '\0'; ++i)
                                    すべてが含まれていれば
                                    true を、いずれかが欠
       if (isdigit(s[i]))
                                    けていれば false を返す
       {
           ++digit;
       else if (islower(s[i]))
           ++lower;
       else if (isupper(s[i]))
           ++upper;
   return (digit && lower && upper);
}
```

```
void DrawCheckerboard(int w, int h, int n, char a, char b)
{
    for (int y = 0; y < h; ++y)
    {
        for (int x = 0; x < w; ++x)
        {
            int y2 = (y / n);
            int x2 = (x / n);
            putchar(((y2 + x2) % 2 == 0) ? a : b);
        }
        putchar('\n');
    }
}</pre>
```

指定された 2 種類の文字 a, b を用いて、全体として幅 w, 高さ h の市松模様を出力する。模様を構成するセルのサイズを n で指定できる。左上の文字は a。

要約して本質的な機能を説明せよ。 I 行ずつ説明した場合は不正解とする。 また、コードに含まれる複数のリスク(未定義動作を引き起こすか、その可能性の ある部分)とその修正案を、コード欄に直接書き込んで説明せよ。

```
#include <stdio.h>
                                        <入力例>
#include <ctype.h>
                                        Waseda 2025
#define NUM_ALPHABETS ('Z' - 'A')
             アルファベットの文字数 26 ではなく 25 になる。
int main(void)
             配列の要素数不足で範囲外アクセスを引き起こす
   int count [NUM_ALPHABETS];
                             配列に初期値を与えていない
   for (;;)
                             ため、要素が不定値。
       char c = (char)getchar(); = { 0 };のように初期化する
       if (c == '\n')
          break;
                          これがないと、アルファベット以
       if (isupper(c))
                          外の入力(空白や数字)に対して
          ++count[c - 'A']; 範囲外アクセスを起こす。
            if (islower(c))
       else
          ++count[c - 'a'];
       }
   }
   for (int i = 0; i < NUM_ALPHABETS; ++i)</pre>
       printf("%c: %d\n", ('A' + i), count[i]);
```

標準入力から改行が入力されるまで文字列を読み込み、小文字と 大文字を区別せず、A,B,Cなど各アルファベットの文字が何回登 要約 場したかを、ABC順に各行に出力する。

Transpose 関数呼び出し後の配列 matrix の状態(要素の並び)を書け。 このプログラムで SwapInt は何回呼び出されるかを書け。 Transpose 関数が SwapInt 関数を呼ぶ回数 C と引数 N の関係を式で表せ。

```
#include <stdio.h>
void SwapInt(int* a, int* b)
{
    int t = *a;
    *a = *b;
    *b = t;
void Transpose(int m[], int N)
    for (int i = 0; i < N; ++i)
        for (int k = (i + 1); k < N; ++k)
            SwapInt(&m[i * N + k], &m[k * N + i]);
int main(void)
    int matrix[] = {
        1, 2, 3, 4,
        5, 6, 7, 8,
        9, 10, 11, 12,
        13, 14, 15, 16
    };
    Transpose(matrix, 4);
   5
        9
           13
                2
                    6
                       10 14 3
                                   7
                                       11 15
                                                4
                                                       12 16
                                                    8
                         N(N-1)
```

6

C =

クラス	学籍番号	氏名
Cプログラミング 入門(電生 2)		

#include <stdio.h>
#include <stdbool.h>

puts("A");

puts("B");

int main(void)

return true;

return false;

bool A()

bool B()

{

{

}

次のコードの実行結果(出力)を書け。

```
#include <stdio.h>
int main(void)
{
    int n = 5;
    n += (n + 15);
    printf("%d\n", n);
    n *= n;
    printf("%d\n", n);
    ++n;
    printf("%d\n", n);
}

25
625
626
```

```
if (A() && A())
                                         puts("C");
                                    }
#include <stdio.h>
                                    if (A() || B() || B())
typedef enum
                                    {
                                         puts("D");
    RED = 100,
                                    }
    GREEN = 200,
                                }
    BLUE,
    YELLOW
                                Α
} Color;
                                Α
int main(void)
{
                                В
    printf("%d\n", BLUE);
    printf("%d\n", YELLOW);
                                D
201
202
```

printf("%d\n", ABS(-500));
printf("%.2f\n", ABS(0.01));
printf("%d\n", ABS(1 - 4));

}

```
#include <stdio.h>
#define COUNTOF(a) ((int)(sizeof(a) / sizeof((a)[0])))
int main(void)
{
   int vals[6] = { 1, 2, 3, 4, 5 };
   for (int i = 0; i < COUNTOF(vals); ++i)
    {
      printf("%d\n", vals[COUNTOF(vals) - 1 - i]);
   }
}</pre>
```

次の式の結果について、何型でどのような値になるかを書け。

	型(int など)	値	
20 / 2LL	long long	IOLL	
pow(2.0, 4.0) <= 10.0	int	0	
-1 ? (-2 ? -3 : -4) : -5	int	-3	
floor(2.6)	double	2.0	
strlen("a\nb\n")	size_t	4	
fmod(2.5, 1.0)	double	0.5	

次のコードを実行し、指定した入力(複数行)を一度に与えた場合の出力を書け。

```
#include <stdio.h>
                                                    入力
#include <ctype.h>
                                               Tokyo 2020
                                               Paris 2024
int main(void)
                                               LA 2028
    int n = 0;
                                               n = 4
    for (;;)
        char c = (char)getchar();
        if (c == '\n')
            break;
        n += (isdigit(c) ? (c - '0') : 0);
    printf("n = %d\n", n);
}
```

#### (参考) ASCII 表

	(多专) ASCII 农										
10進	16進	文字	10進	16進	文字	10進	16進	文字	10進	16進	文字
0	00	∖0 (ヌル文字)	32	20	space	64	40	0	96	60	`
1	01	SOH	33	21	!	65	41	Α	97	61	a
2	02	STX	34	22	"	66	42	В	98	62	b
3	03	ETX	35	23	#	67	43	С	99	63	С
4	04	EOT	36	24	\$	68	44	D	100	64	d
5	05	ENQ	37	25	%	69	45	E	101	65	е
6	06	ACK	38	26	&	70	46	F	102	66	f
7	07	\a (ベル)	39	27	-	71	47	G	103	67	g
8	08	<b>\b</b> (バックスペース)	40	28	(	72	48	Н	104	68	h
9	09	<b>\t</b> (水平タブ)	41	29	)	73	49	I	105	69	i
10	0A	<b>\n</b> (改行)	42	2A	*	74	4A	J	106	6A	j
11	0B	\v (垂直タブ)	43	2B	+	75	4B	К	107	6B	k
12	0C	<b>\f</b> (改ページ)	44	2C	,	76	4C	Г	108	6C	l
13	0D	<b>\r</b> (復帰)	45	2D	-	77	4D	М	109	6D	m
14	0E	S0	46	2E		78	4E	N	110	6E	n
15	0F	SI	47	2F	/	79	4F	0	111	6F	0
16	10	DLE	48	30	0	80	50	Р	112	70	р
17	11	DC1	49	31	1	81	51	Q	113	71	q
18	12	DC2	50	32	2	82	52	R	114	72	r
19	13	DC3	51	33	3	83	53	S	115	73	S
20	14	DC4	52	34	4	84	54	T	116	74	t
21	15	NAK	53	35	5	85	55	C	117	75	u
22	16	SYN	54	36	6	86	56	٧	118	76	v
23	17	ETB	55	37	7	87	57	W	119	77	W
24	18	CAN	56	38	8	88	58	X	120	78	X
25	19	EM	57	39	9	89	59	Υ	121	79	У
26	1A	SUB	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	59	3B	i	91	5B	[	123	7B	{
28	1C	FS	60	3C	<	92	5C	\	124	7C	
29	1D	GS	61	3D	=	93	5D	]	125	7D	}
30	1E	RS	62	3E	>	94	5E	^	126	7E	2
31	1F	US	63	3F	?	95	5F	_	127	7F	DEL

## 次の説明に合う関数を最下段の選択肢からすべて選んで書け。

渡した文字が小文字であるかを返す	isdigit
渡した double 型の値の絶対値を返す	fabs
整数値を渡すと、それに対応する ASCII 文字を出力する	putchar, printf
ヌル終端の前までにある char 型の要素数を返す	strlen
乱数生成器のシード値を、指定した整数に変更する	srand

引数が 0 個である getchar, rand <math.h> の関数 fabs, ceil, floor, pow, hypot 戻り値が double 型 fabs, ceil, floor, pow, hypot

puts printf scanf getchar putchar isdigit islower isalnum rand srand abs fabs ceil floor pow hypot strlen

誤っている部分があれば下線を引き、それを置き換える正しい記述を書け。 誤っている部分が無い場合は「正しい」と書け。

構造体は配列をメンバ変数として持つことはできない

#### 持つことができる

配列 int n[10]; の先頭要素は n[1] で、末尾の要素は n[10]

n[0] で、末尾の要素は n[9]

変数nが0以上10以下であるかは、0くnく10という式で判定する

 $0 \le n \& n \le 10$ 

scanfで使う double 型用の書式指定子は %1f (パーセントいちエフ)

%1f(パーセントエルエフ)

マクロ名には小文字を使わないのが一般的である

#### 正しい

# 要約して本質的な機能を説明せよ。 | 行ずつ説明した場合は不正解とする。

```
与えられた文字列に「数
bool Check(const char s[])
                                    字」「小文字」「大文
   int digit = 0, lower = 0, upper = 0; |字」がそれぞれ | つ以上
                                    含まれているかを判定し
   for (int i = 0; s[i] != '\0'; ++i)
                                    すべてが含まれていれば
                                    true を、いずれかが欠
       if (isdigit(s[i]))
                                    けていれば false を返す
       {
           ++digit;
       else if (islower(s[i]))
           ++lower;
       else if (isupper(s[i]))
           ++upper;
   return (digit && lower && upper);
}
```

```
void DrawCheckerboard(int w, int h, int n, char a, char b)
{
    for (int y = 0; y < h; ++y)
    {
        for (int x = 0; x < w; ++x)
        {
            int y2 = (y / n);
            int x2 = (x / n);
            putchar(((y2 + x2) % 2 == 0) ? a : b);
        }
        putchar('\n');
    }
}</pre>
```

指定された 2 種類の文字 a,b を用いて、全体として幅 w,高さ b の市松模様を出力する。模様を構成するセルのサイズを b で指定できる。左上の文字は a。

要約して本質的な機能を説明せよ。 I 行ずつ説明した場合は不正解とする。 また、コードに含まれる複数のリスク(未定義動作を引き起こすか、その可能性の ある部分)とその修正案を、コード欄に直接書き込んで説明せよ。

```
#include <stdio.h>
                                        <入力例>
#include <ctype.h>
                                        Waseda 2025
#define NUM_ALPHABETS ('Z' - 'A')
             アルファベットの文字数 26 ではなく 25 になる。
int main(void)
             配列の要素数不足で範囲外アクセスを引き起こす
   int count [NUM_ALPHABETS];
                             配列に初期値を与えていない
   for (;;)
                             ため、要素が不定値。
       char c = (char)getchar(); = { 0 };のように初期化する
       if (c == '\n')
          break;
                          これがないと、アルファベット以
       if (isupper(c))
                          外の入力(空白や数字)に対して
          ++count[c - 'A']; 範囲外アクセスを起こす。
            if (islower(c))
       else
          ++count[c - 'a'];
       }
   }
   for (int i = 0; i < NUM_ALPHABETS; ++i)</pre>
       printf("%c: %d\n", ('A' + i), count[i]);
```

標準入力から改行が入力されるまで文字列を読み込み、小文字と 大文字を区別せず、A,B,Cなど各アルファベットの文字が何回登 要約 場したかを、ABC順に各行に出力する。

Transpose 関数呼び出し後の配列 matrix の状態(要素の並び)を書け。 このプログラムで SwapInt は何回呼び出されるかを書け。 Transpose 関数が SwapInt 関数を呼ぶ回数 C と引数 N の関係を式で表せ。

```
#include <stdio.h>
void SwapInt(int* a, int* b)
{
    int t = *a;
    *a = *b;
    *b = t;
void Transpose(int m[], int N)
    for (int i = 0; i < N; ++i)
        for (int k = (i + 1); k < N; ++k)
            SwapInt(&m[i * N + k], &m[k * N + i]);
int main(void)
    int matrix[] = {
        1, 2, 3, 4,
        5, 6, 7, 8,
        9, 10, 11, 12,
        13, 14, 15, 16
    };
    Transpose(matrix, 4);
   5
        9
           13
                2
                    6
                       10 14 3
                                   7
                                       11 15
                                                4
                                                       12 16
                                                    8
                         N(N-1)
```

6

C =