

1. 答え合わせ
2. 配列補足
3. 関数補足
4. 反復

問2 絶対値

```
1 #include<stdio.h>
2 int abs(int x){
3     /* */
4 }
5 int main(void){
6     int n;
7     scanf("%d", &n);
8     printf("%d\n", abs(n));
9     return 0;
10 }
```

問2 答

```
1 int abs(int x){
2     if( x < 0 ){
3         x = x * (-1);
4     }
5     return x;
6 }
```

変数/配列 宣言

int num; &num



int arr[3]; &arr[0] &arr[1] &arr[2]



変数/配列 代入

```
arr[0] = 8;    arr[1] = 4;  
scanf("%d", &arr[2]);
```

// 入力を待ち 2が入力されたとする

&arr[0]

&arr[1]

&arr[2]

arr[0] = 8

arr[0] = 4

arr[0] = 2

変数/配列 出力

```
printf(“%d\n”, 16 ); // 16
```

```
printf(“%d\n”, arr[0] ); // 8
```

&arr[0]

&arr[1]

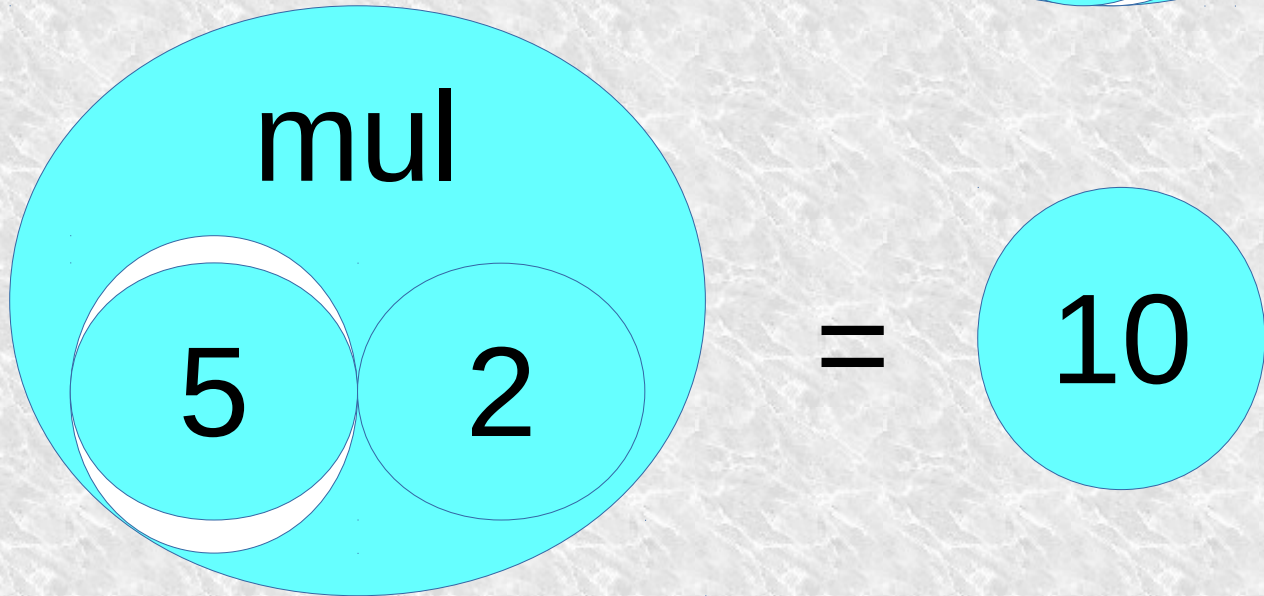
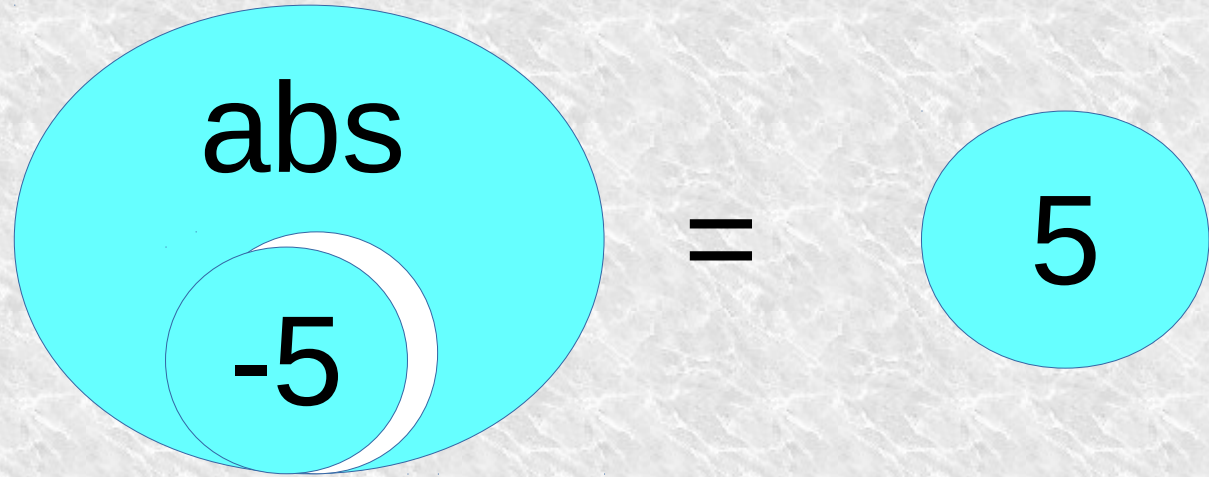
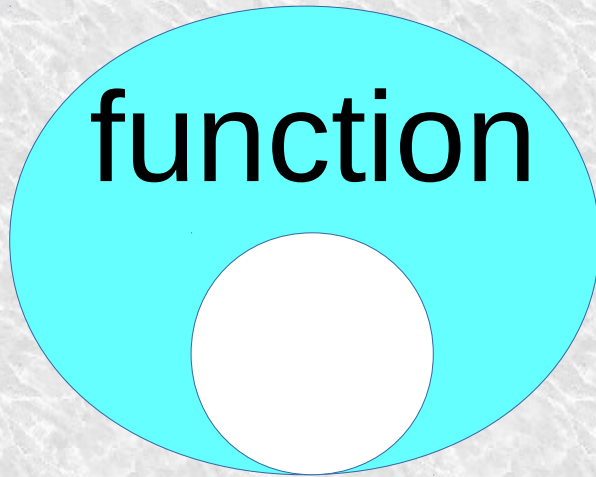
&arr[2]

arr[0] = 8

arr[0] = 4

arr[0] = 2

関数



反復

```
1#include<stdio.h>
2int main(void){
3    for(int i = 0; i < 10; i = i + 1){
4        printf("%d\n", i);
5    }
6    return 0;
7}
```


問3 べき乗

正整数 x, y を引数にとって
 x の y 乗を返す関数を定義

問3 答

```
1 int power(int x, int y){
2     int res = 1;
3     for(int i = 0; i < y; i = i + 1){
4         res = res * x;
5     }
6     return res;
7 }
```

GNU Make

```
C = gcc  
CFLAGS = -std=c99 -Wall  
SOURCE = $(wildcard *.c)  
all: $(basename $(SOURCE))
```

```
%.c  
$(C) -o $@ $^ $(CFLAGS)
```

Makefile

\$ make -k

gcc -o hoge hoge.c -std=c99 -Wall

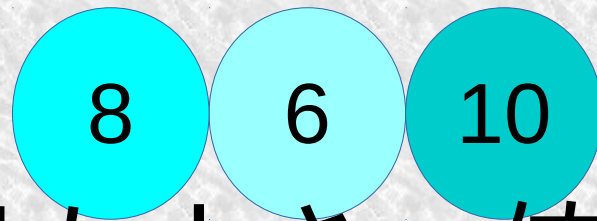
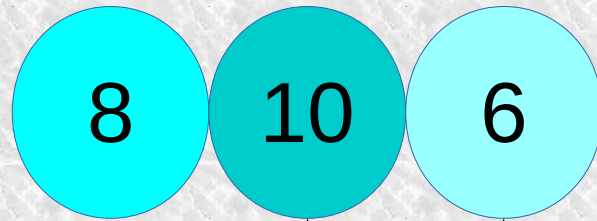
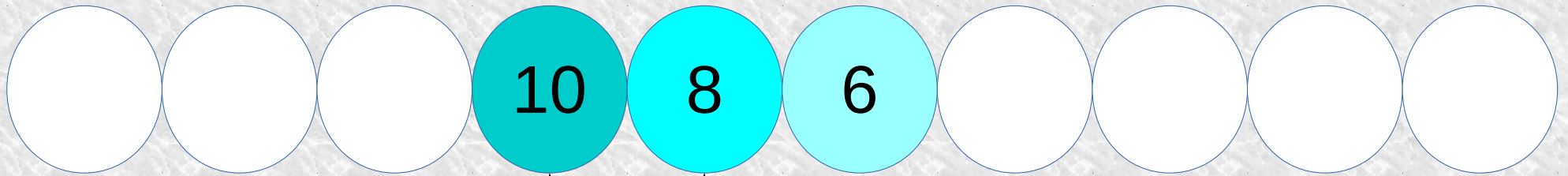
問3.5 FizzBuzz

1から100までの整数について
15の倍数ならFizzBuzz
3の倍数ならFizz
5の倍数ならBuzz
いずれでもないならその数字を
出力するプログラム

問4 昇順ソート

```
1#include<stdio.h>
2void sort(int array[], int size){
3    /* */
4}
5int main(void){
6    int array[10];
7    for(int i = 0; i < 10; i = i + 1){
8        scanf("%d", &array[i]);
9    }
10   sort(array, 10);
11   for(int i = 0; i < 10; i = i + 1){
12       printf("%d\n", array[i]);
13   }
14   return 0;
15}
```

問4 ヒント



繰り返すと最も大きい値が右端にくる

問4 答

```
1 void sort(int array[], int size){
2     for(int i = 0; i < size; i = i+1){
3         for(int k = 1; k < size - i; k = k+1){
4             if( array[k] < array[k-1] ){
5                 int tmp = array[k];
6                 array[k] = array[k-1];
7                 array[k-1] = tmp;
8             }
9         }
10    }
11 }
```