

## Algoritme dan Pemrograman

- Kuliah #13
- Operator Bitwise
  - Tipe Enumerasi
  - Manipulasi File

### Bit

- Singkatan dari binary digit (digit biner), yaitu suatu nilai bilangan yang direpresentasikan sebagai bilangan biner (basis 2)
- Contoh:
  - bilangan bulat desimal 156 dapat dituliskan sebagai 10011100 (8 bit)
  - Bilangan bulat desimal 20 dapat dituliskan sebagai 00010100 (8 bit)

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Bitwise operator

- Operator yang digunakan untuk memanipulasi bit-bit dari operand-operand tipe data integral (char, short, int, long).
- Jenis operator:

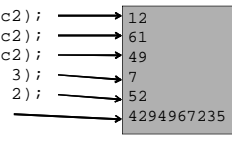
~	one's complement (bitwise NOT)
&	bitwise AND
	bitwise inclusive OR
^	bitwise exclusive OR
<<	left shift
>>	right shift

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Apa output program berikut?

```
#include <stdio.h>

main()
{
    unsigned int c1=60;
    unsigned int c2=13;
    printf("%u\n", c1 & c2);
    printf("%u\n", c1 | c2);
    printf("%u\n", c1 ^ c2);
    printf("%u\n", c1 >> 3);
    printf("%u\n", c2 << 2);
    printf("%u\n", ~c1);
    return 0;
}
```



DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Fungsi menampilkan n bits dari suatu bilangan desimal

```
void bits (unsigned value, unsigned n) {
    unsigned int c, mask = 1 << (n-1);

    for ( c = 1 ; c <= n ; c++ ) {
        printf("%c", ( value & mask ? '1' : '0' ));
        mask = mask >> 1;
    }
}
```


Apa yang akan dicetak dari pemanggilan bits(20, 4) ?

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Apa output program berikut?

```
#include <stdio.h>
... // fungsi bits

main()
{
    unsigned int c1=60, c2=13, c3, c4, c5;
    c3 = c1 & c2;
    c4 = c1 | c2;
    c5 = ~c1;
    bits(c3, 8); printf("\n");
    bits(c4, 8); printf("\n");
    bits(c5, 8); printf("\n");
    return 0;
}
```



DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## TIPE DATA ENUMERASI

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Tipe data ENUMERASI

- Tipe enumerasi adalah sekumpulan konstanta integer yang direpresentasikan dengan identifikasi tertentu.
- Mirip dengan himpunan dalam matematika, dimana setiap anggotanya berupa simbol yang memiliki nilai deret.
- Menggunakan kata kunci enum.
- Nilai dalam enum dimulai dari 0, tetapi dapat diubah dengan nilai lainnya, dan menaik dengan penambahan 1 untuk nilai selanjutnya.

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Tipe data ENUMERASI Contoh

```
enum bulan {JAN = 1, FEB, MAR, APR, MAY,
            JUN, JUL, AUG, SEP, OCT, NOV, DEC};
```

- Menciptakan tipe data baru bernama enum bulan.
- Secara otomatis menunjukkan deret nilai 1 untuk JAN hingga 12 untuk DEC.

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Apa output program berikut?

```
#include <stdio.h>

enum bulan {JAN=1, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP,
            OCT, NOV, DEC};

main() {
    enum bulan Bulan;

    const char *namaBulan[] = {"", "January", "February",
                                "March", "April", "May", "June", "July", "August",
                                "September", "October", "November", "December"};

    for (Bulan=JAN; Bulan<=DEC; Bulan++) {
        printf("%2d %s\n", Bulan, namaBulan[Bulan]);
    }

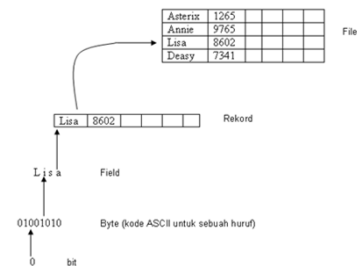
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## FILE

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

### Hirarki data



DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## File dalam C

- C merepresentasikan file secara sederhana sebagai n bytes stream.
- Saat membuka file berarti memberi nilai pointer ke struktur FILE dalam pustaka baku <stdio.h>.
- Tiga file dan stream terkait secara otomatis dibuka pada saat program mulai diproses, yaitu standard input, standard output, dan standard error, dimanipulasi dengan pointer file stdin, stdout, dan stderr.

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Beberapa fungsi dalam pustaka (library) C

```
FILE *fopen(const char *namafile,
            const char *modus);
int fgetc (FILE *stream);
char *fgets (char *s, int n,
            FILE *stream);
int fputc (int c, FILE *stream);
int fputs (const char *s, FILE *stream);
int fclose (FILE *stream);
int fprintf (FILE *stream,
            const char *format, ...);
int fscanf (FILE *stream,
            const char *format, ...);
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Membuka file

```
FILE *fopen(const char *namafile, const char *modus);
```

- Pointer ke file
- String untuk modus:
  - r Open a text file for reading
  - w Create a text file for writing
  - a Append to a text file
  - rb Open a binary file for reading
  - wb Open a binary file for writing
  - ab Append to a binary file
  - r+ Open a text file for read/write
  - w+ Create a text file for read/write
  - a+ Append or create a text file for read/write
  - r+b Open a binary file for read/write
  - w+b Create a binary file for read/write
  - a+b Append a binary file for read/write

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Text File Functions

- Menulis/membaca string ke/dari file:

```
int fputs(char *str, FILE *fp);
char *fgets(char *str, int num, FILE *fp);
```

- Menulis/membaca dengan format tertentu

```
int fprintf(FILE *fp, char *control-string, ...);
int fscanf(FILE *fp, char *control-string ...);
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Contoh Membaca isi file data.txt

```
#include <stdio.h>
main() {
    FILE *in;
    char line[200];
    int uts, uas;
    if (( in = fopen("data.txt" , "r")) == NULL)
        printf ("File tidak bisa dibaca\n");
    else {
        while (!feof(in)) {
            fgets(line, sizeof(line), in);
            printf("%s", line);
        }
        fclose(in);
        return 0;
    }
}
```

```
G64084120 60 80
G64084126 70 50
G64084123 90 80
G64084125 40 30
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Contoh Membaca isi file data.txt

```
#include <stdio.h>
main() {
    FILE *in;
    char nim[10];
    int uts, uas;
    if (( in = fopen("data.txt" , "r")) == NULL)
        printf ("File tidak bisa dibaca\n");
    else {
        while (!feof(in)) {
            fscanf(in, "%s %d %d", &nim, &uts, &uas);
            printf("%s %d %d\n", nim, uts, uas);
        }
        fclose(in);
        return 0;
    }
}
```

```
G64084120 60 80
G64084126 70 50
G64084123 90 80
G64084125 40 30
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Binary File Functions

```
size_t fread(void *buffer, size_t size,
             size_t num, FILE *fp);

size_t fwrite(void *buffer, size_t size,
              size_t num, FILE *fp);
```

- buffer** pointer to the array of elements to be written
- size** size in bytes of each element to be written
- num** number of elements, each one with a size of size bytes
- fp** file stream

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Contoh Menulis data ke file biner

```
#include <stdio.h>
struct mhs {
    char nim[10];
    int uts, uas;
};
typedef struct mhs MHS;
main() {
    FILE *out;
    MHS list[2]={{ "G64084120", 30, 50},
                 {"G64084122", 60, 70}};
    out = fopen("myfile.dat", "wb");
    fwrite(list, 1, sizeof(list), out);
    fclose(out);
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Contoh Membaca data dari file biner

```
#include <stdio.h>
struct mhs {
    char nim[10];
    int uts, uas;
};
typedef struct mhs MHS;
main() {
    FILE *in;
    MHS list[2];
    int i;
    in = fopen("myfile.dat", "rb");
    fread(list, 1, sizeof(list), in);
    for (i=0; i<2; i++)
        printf("%s %d %d\n", list[i].nim, list[i].uts,
              list[i].uas);
    fclose(in);
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Latihan #1

- Ubah program membaca isi file data.txt pada slide sebelumnya sehingga ditampilkan isi data terurut berdasarkan nim.
- Gunakan tipe data struct untuk menyimpan data nim, uts, dan uas.
- Gunakan algoritme bubble sort untuk mengurutkan data berdasarkan nim.

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR

## Latihan #2

- Tersedia data pada file fpb.in yang berisi n buah pasangan bilangan bulat seperti berikut (tidak harus 3 baris):
 

```
28 12
17 36
9 21
```
- Buat program menentukan FPB dari setiap pasangan bilangan yang ada, dan simpan hasilnya pada file fpb.out seperti berikut:
 

```
4
1
3
```

DEPARTEMEN ILMU KOMPUTER  
INSTITUT PERTANIAN BOGOR