

# Informatyka 2 (ES1F2012)

---

Politechnika Białostocka - Wydział Elektryczny  
Elektrotechnika, semestr II, studia stacjonarne I stopnia  
Rok akademicki 2022/2023

**Pracownia nr 8 (20.04.2023)**

dr inż. Jarosław Forenc

# Operacje na plikach

- Strumień wiąże się z plikiem za pomocą **otwarcia**, zaś połączenie to jest przerywane przez **zamknięcie** strumienia
- Operacje związane z przetwarzaniem pliku zazwyczaj składają się z trzech części

## 1. Otwarcie pliku (strumienia):

- funkcje: **fopen()**

## 2. Operacje na pliku (strumieniu), np. czytanie, pisanie:

- funkcje dla plików tekstowych: **fprintf(), fscanf(), fgetc(), fputc(), fgets(), fputs()...**

- funkcje dla plików binarnych: **fread(), fwrite(), ...**

## 3. Zamknięcie pliku (strumienia):

- funkcja: **fclose()**

## Otwarcie pliku - fopen()

```
FILE* fopen(const char *fname, const char *mode);
```

- **fname** - nazwa pliku, może zawierać całą ścieżkę dostępu do pliku
- **mode** - tryb otwarcia:
  - **"r"** - odczyt
  - **"w"** - zapis - jeśli pliku nie ma to zostanie on utworzony, jeśli plik istnieje, to jego poprzednia zawartość zostanie usunięta
  - **"a"** - zapis (dopisywanie) - dopisywanie danych na końcu istniejącego pliku, jeśli pliku nie ma to zostanie utworzony
  - **"t"** - otwarcie w trybie tekstowym (domyślnie)
  - **"b"** - otwarcie w trybie binarnym
- **fopen()** zwraca wskaźnik na strukturę **FILE** skojarzoną z otwartym plikiem lub **NULL**, gdy otwarcie nie powiodło się

## Otwarcie pliku - fopen()

- Otwarcie pliku w trybie tekstowym, tylko odczyt

```
FILE *fp;  
fp = fopen("dane.txt", "r");
```

- Otwarcie pliku w trybie binarnym, tylko zapis

```
fp = fopen("c:\\baza\\data.bin", "wb");
```

- Otwarcie pliku w trybie tekstowym, tylko zapis

```
fp = fopen("wynik.txt", "wt");
```

## Zamknięcie pliku - fclose()

```
int fclose(FILE *fp);
```

- zamyka plik wskazywany przez `fp`

```
#include <stdio.h>

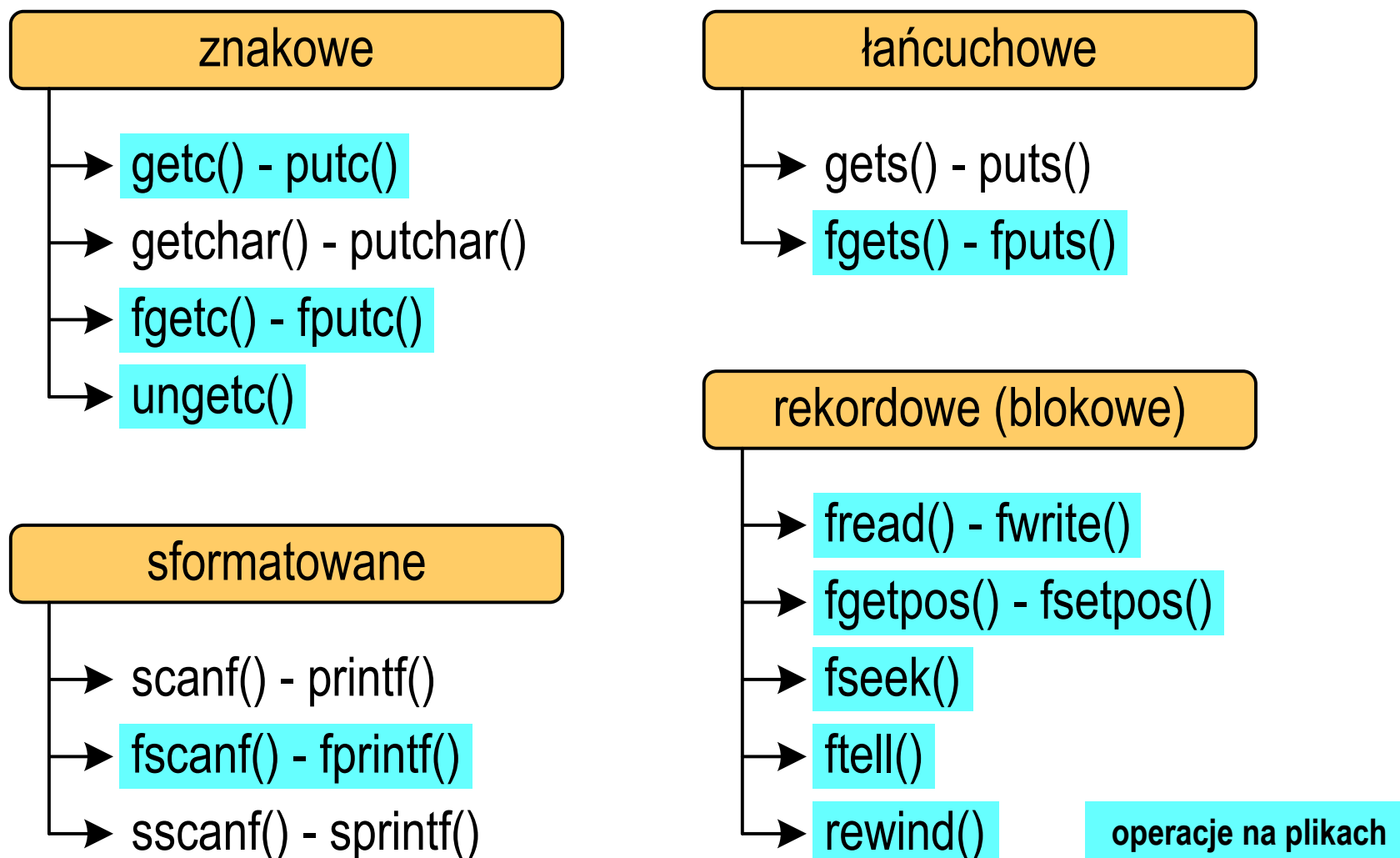
int main(void) {
    FILE *fp;

    fp = fopen("plik.txt", "w");
    if (fp == NULL) {
        printf("Bład otwarcia pliku.\n");
        return (-1);
    }

    /* przetwarzanie pliku */

    fclose(fp);
    return 0;
}
```

# Typy standardowych operacji wejścia-wyjścia (pliki)



## Znakowe operacje wejścia-wyjścia

```
int putc(int znak, FILE *fp);
```

```
int fputc(int znak, FILE *fp);
```

- wpisuje **znak** do otwartego strumienia **fp**

```
FILE *fp; int zn = 'a';  
putc(zn, fp);           // do pliku  
fputc(zn, fp);         // do pliku
```

## Znakowe operacje wejścia-wyjścia

```
int getc(FILE *fp);
```

```
int fgetc(FILE *fp);
```

- pobiera (czyta) jeden znak ze strumienia `fp` i zwraca jego kod (jako `int`) lub `EOF` (gdy napotkano koniec pliku)

```
FILE *fp; int zn;  
zn = getc(fp); // z pliku  
zn = fgetc(fp); // z pliku
```



## Łańcuchowe operacje wejścia-wyjścia

```
int fputs(const char *buf, FILE *fp);
```

- wpisuje znaki z tablicy `buf` do otwartego strumienia `fp`;  
nie dołącza znaku końca wiersza `'\n'`

```
FILE *fp; char txt[20] = "Witaj swiecie";  
fputs(txt, fp); // do pliku
```

```
char* fgets(char *buf, int max, FILE *fp);
```

- czyta znaki z otwartego strumienia `fp` i zapisuje je w tablicy `buf`;  
przerywa pobieranie znaków po odczytaniu `'\n'` lub `max-1` znaków;  
zwraca `NULL` po napotkaniu końca pliku

```
FILE *fp; char txt[20];  
fgets(txt, 20, fp); // z pliku
```

## Sformatowane operacje wejścia-wyjścia

```
int fprintf(FILE *fp, const char *format, ...);
```

- wyprowadza dane do otwartego strumienia `fp` (najczęściej pliku)

```
FILE *fp; char txt[30];  
fprintf(fp, "Witaj swiecie"); // do pliku
```

```
int fscanf(FILE *fp, const char *format, ...)
```

- czyta dane z otwartego strumienia `fp` (najczęściej pliku)

```
FILE *fp; int x; float y;  
fscanf(fp, "%d %f", &x, &y); // z pliku
```

## Wykrycie końca pliku tekstowego

Funkcje	Metoda
getc(), fgetc()	zwracana wartość: EOF
fgets()	zwracana wartość: NULL
fscanf()	wywołanie funkcji: feof()

```
int feof(FILE *fp);
```

- zwraca wartość różną od zera, jeśli podczas ostatniej operacji odczytu pliku wskazywanego przez `fp` został wykryty jego koniec; w przeciwnym razie zwraca wartość `0` (zero)

## Przykład: wyświetlenie pliku tekstowego

```
#include <stdio.h>

int main(void)
{
    FILE *fp;
    int znak;

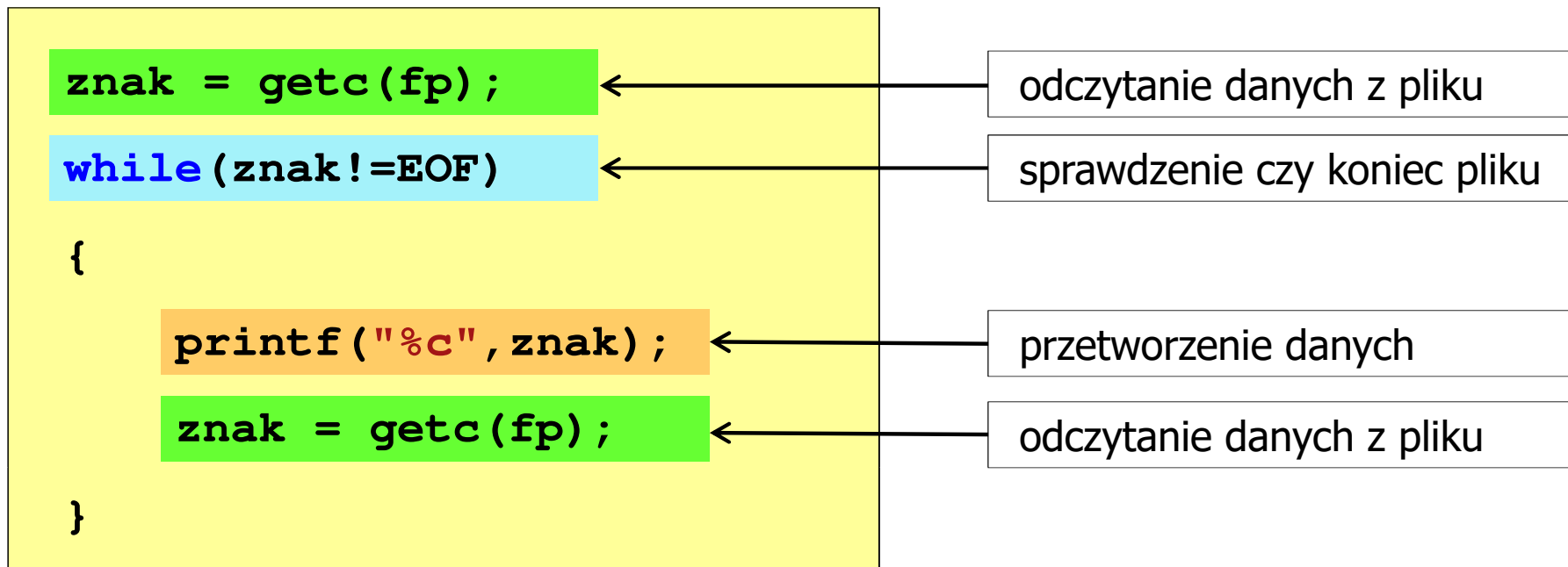
    fp = fopen("test.txt", "r");

    znak = getc(fp);
    while (znak != EOF)
    {
        printf("%c", znak);
        znak = getc(fp);
    }

    fclose(fp);
    return 0;
}
```

# Schemat przetwarzania pliku

- Typowy schemat odczytywania danych z pliku



- Krótszy zapis:

```
while ( (znak=getc(fp)) !=EOF)  
    printf ("%c", znak);
```