



MAGISTER TEKNIK INFORMATIKA

MATRIKULASI – STRUKTUR DATA

LAPORAN PRAKTIKUM

Nama : Rakha Asyrofi
NRP : 05111950010038

Mata Kuliah : Struktur Data
Dosen : Dr. Umi Laili Yuhana S.Kom., M.Sc

Tugas Praktikum 2

1. Divide by Zero (Pembagian dengan nol)

```
#include <stdio.h>

int main()
{
    int a, b;
    scanf("%d%d", &a, &b);
    printf("%d", a/b);
    return 0;
}
```

Dalam statement disini bisa terjadi error jika bilangan dibagi dengan 0, disebabkan karena modul compiler pada c ini tidak dapat menghasilkan angka infinite (tidak terbatas) dapat dilihat pada hasil output pada program sebagai berikut:

```
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc1
4
2
2
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc1
4
0
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> █
```

Maka solusi yang ditawarkan bila dibagi 0 dibuatkan statement lain seperti berikut, sehingga dinyatakan sebagai kondisi lainnya sebagai string.

```
#include <stdio.h>

int main()
{
    int a, b;
    scanf("%d %d", &a, &b);
    if (b!=0)
    {
        printf("%d", a/b);
    }
    else
    {
        printf("anda membagi dengan 0\n");
    }
    return 0;
}
```

Maka hasilnya sebagai berikut, nilai tertulis bila dibagi 0 adalah string karate tertentu.

```
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc1
4
2
2
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc1
4
0
anda membagi dengan 0
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> █
```

2. Integer/Integer

```
#include <stdio.h>

int main()
{
    int a, b;
    double c;
    scanf("%d %d", &a, &b);
    if (b!=0)
    {
        c = a/b;
        printf("%lf", c);
    }
    else
    {
        printf("anda membagi dengan 0\n");
    }

    return 0;
}
```

Dimana jika membagi antara 2 integer yang seharusnya nilai pembagian tersebut kita rubah ke variabel float/double maka nilainya tidak sesuai dengan harapan. Bisa dilihat dari hasil sebagai berikut:

```
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc2
4
2
2.000000
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc2
5
2
2.000000
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> █
```

Maka solusi ditawarkan yaitu dengan penggunaan type cast pada proses formula pembagiannya sebagai berikut. Sehingga hasil bagi dari integer kita konversikan ke nilai bertipe double.

```
#include <stdio.h>

int main()
{
    int a, b;
    double c;
    scanf("%d %d", &a, &b);
    if (b!=0)
    {
        c = (double)a/b;
        printf("%lf", c);
    }
    else
    {
        printf("anda membagi dengan 0\n");
    }

    return 0;
}
```

Sehingga hasil bila 2 variabel integer saling dibagi menghasilkan nilai hasil dari konversi int ke double.



MAGISTER TEKNIK INFORMATIKA

MATRIKULASI – STRUKTUR DATA

```
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc2
4
2
2.000000
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> .\Rakha_Asyrofi_05111950010038_perc2
5
2
2.500000
PS C:\Users\Asus\Desktop\tugas\Struktur Data(c)\strukturData\modul2> |
```

3. Overflow

```
#include<stdio.h>
int main()
{
    int i = 2147483647;
    printf("%d", i);
    return 0;
}
```

Dimana dalam statement ini variable yang dideklarasikan yaitu int dengan range antara -2.147.483.648 sampai 2.147.483.647 sehingga hasil didapatkan sebagai berikut:

```
[Running] cd "c:\Users\Asus\Desktop\tugas\Str
2147483647
[Done] exited with code=0 in 1.351 seconds
```

Sedangkan saat kita tambahkan nilai dari variable itu 1, sehingga menyebabkan overflow kelebihan dari range yang ditentukan. Maka nilai dari keluaran tersebut berupa nilai range awal dari variable int tersebut, berikut statement yang dikeluarkan:

```
#include<stdio.h>
int main()
{
    int i = 2147483648;
    printf("%d", i);
    return 0;
}
```

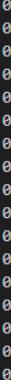
Maka hasil dari perulangan tersebut menjadi sebagai berikut:

```
[Running] cd "c:\Users\Asus\Desktop\tugas\Str
-2147483648
[Done] exited with code=0 in 1.389 seconds
```

4. Infinite Loop

```
#include <stdio.h>

int main()
{
    int i;
    i = 0;
    while (i < 5)
    {
        printf("%d\n", i);
    }
    return 0;
}
```



Maka ada solusi dari permasalahan tersebut dengan menggunakan pemenuhan statemen dari perulangan while bila kondisi yang ditentukan sudah terpenuhi. Misalnya dari contoh program sebagai berikut bila variable I ditambahkan 1 setiap increment dari statement sebelumnya.

```
[Running] cd "c:\Users\Asus\Desktop\tugas\Str"
1
2
3
4
5

[Done] exited with code=0 in 1.228 seconds
```

```
#include <stdio.h>

int main()
{
```



MAGISTER TEKNIK INFORMATIKA

MATRIKULASI – STRUKTUR DATA

```
int i;  
i = 0;  
  
do  
{  
    printf("%d\n", i+1);  
    i++;  
}  
while (i < 5);  
  
return 0;  
}
```

Prinsipnya hampir sama sebelumnya dilakukan proses perulangan dilakukan seminimal mungkin untuk melakukan pengecekan terhadap kondisi yang telah dilakukan. Hasil yang didapatkan pun sama seperti diatas sebelumnya. Sebagai berikut:

```
[Running] cd "c:\Users\Asus\Desktop\tugas\Str  
1  
2  
3  
4  
5  
  
[Done] exited with code=0 in 1.228 seconds
```

4.3 Perulangan dengan for

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

Dalam perulangan ini dilakukan sama halnya dengan perulangan while yang ditentukan kondisi tertentu untuk menyelesaikan programnya menggunakan increament.

```
[Running] cd "c:\Users\Asus\Desktop\tugas\Str  
1  
2  
3  
4  
5  
  
[Done] exited with code=0 in 1.228 seconds
```

5. Konsep Dasar Array

Array merupakan kumpulan data yang sama dengan nama variable maupun statement yang sama. Dengan menggunakan array memakai variable nama yang sama. Untuk membedakan antar array digunakan variabel - variabel lain yang digunakan untuk menentukan indexnya.

a. Array Satu Dimensi

Array ini merupakan kumpulan dari beberapa data yang memiliki variabel maupun nama yang sama namun sebagai pembeda menggunakan index. Berikut ini adalah penggunaan array 1 dimensi.

Dimana dalam array ini dideklarasikan seperti halnya variabel biasa, tetapi dengan kapasitas dalam kurung siku, isi array yang bisa diakses adalah antara 0 sampai kapasitas -1, jika mendeklarasikan array berkapasitas 10, indeks yang bisa diakses hanya indeks 0 hingga 9. Kemudian assign & akses isi array dengan variabel nama yang sama, tetapi menggunakan index.

```
#include <stdio.h>

int main()
{
    int a[10];
    int b[10];
    a[0] = 100;
    a[1] = 20;
    printf("%d %d\n", a[0], a[1]);
    return 0;
}
```

```
[Running] cd "c:\Users\Asus\Desktop\tugas\Str"
100 20
[Done] exited with code=0 in 1.054 seconds
```

Mengapa kita memerlukan array? Karena bila kita membutuhkan inputan data, lebih dari 100 data, kita tidak perlu repot-repot mendeklarasikan variabel a1 hingga a100, kita cukup menuliskan 1 variabel a dengan tipe array berkapasitas 1000.

Contoh program tanpa array:

```
#include <stdio.h>

int main()
{
    int a, b, c, d, e;
    scanf("%d %d %d %d %d", &a, &b, &c, &d, &e);
    printf("Bilangan ke-1 adalah %d\n", a);
    printf("Bilangan ke-2 adalah %d\n", b);
    printf("Bilangan ke-3 adalah %d\n", c);
    printf("Bilangan ke-4 adalah %d\n", d);
    printf("Bilangan ke-5 adalah %d\n", e);
    return 0;
}
```

```
PS C:\Users\Asus\Desktop\t
1 2 3 4 5
Bilangan ke-1 adalah 1
Bilangan ke-2 adalah 2
Bilangan ke-3 adalah 3
Bilangan ke-4 adalah 4
Bilangan ke-5 adalah 5
PS C:\Users\Asus\Desktop\t
```

```
#include <stdio.h>

int main()
{
    int a[5], i;
    for (i = 0; i < 5; i++)
    {
        scanf("%d", &a[i]);
    }
    for (i = 0; i < 5; i++)
    {
        printf("Bilangan ke-%d adalah %d\n", i+1, a[i]);
    }
    return 0;
}
```



MAGISTER TEKNIK INFORMATIKA

MATRIKULASI – STRUKTUR DATA

b. Array Multidimensi

Dalam array menampung lebih dari satu array. Dalam multidimensi memiliki 2 atau lebih index untuk mengakses seluruh elemen dalam array tersebut.

Berikut adalah contohnya program dengan array 2 dimensi.

```
#include <stdio.h>

int main()
{
    int matriks[10][10];
    matriks[2][3] = 100;
    printf("%d\n", matriks[2][3]);
    return 0;
}
```

```
[Running] cd "c:\Users\Asus\Desktop\tugas\St
100

[Done] exited with code=0 in 1.126 seconds
```

6. Konsep Dasar String

String merupakan tipe data yang menyimpan kumpulan karakter/symbol. Dalam bahasa pemrograman C, suatu string dideklarasikan sebagai array yang bertipe char

Berikut merupakan contoh inputan string dari user, kita dapatkan bila menggunakan scanf atau gets. Perintah scanf akan membaca inputan string dari user dan berhenti ketika ada spasi, enter ataupun interupsi dari pengguna. Sedangkan gets akan membaca satu baris kumpulan karakter hingga enter atau interupsi dari pengguna.

Contoh penggunaan scanf untuk membaca string:

```
#include <stdio.h>

int main()
{
    char arr[10];

    while (1)
    {
        scanf("%s", arr);
        printf("-- %s\n", arr);
    }
    return 0;
}
```

```
PS C:\Users\Asus\Desktop\t
saya
-- saya
siap
-- siap
menjadi
-- menjadi
luar biasa
-- luar
-- biasa
```

Contoh penggunaan gets untuk membaca string:

```
#include <stdio.h>

int main()
{
    char arr[10];

    while (1)
    {
        gets(arr);
        printf("-- %s\n", arr);
    }
    return 0;
}
```

```
PS C:\Users\Asus\Desktop\t
12 23 34 45
-- 12 23 34 45
siap mantap
-- siap mantap
PS C:\Users\Asus\Desktop\t
```

String yang dibaca dengan menggunakan scanf atau gets akan secara otomatis memiliki null character di akhir stringnya.

Fungsi-Fungsi String

Dalam terdapat library yang dibuat dengan tujuan memudahkan pengguna dalam mengolah string. Library tersebut tersimpan dalam string.h, oleh karena itu untuk mengakses library ini, diperlukan tambahan include yaitu: `#include <string.h>`

Berikut ini adalah fungsi-fungsi yang dibagi berdasarkan kegunaanya dalam mengolah sebuah string. Yaitu sebagai berikut:

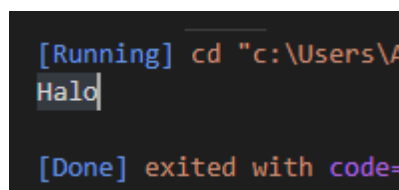
1. Copying
 - a. **Memcpy (copy block of memory)**
 - b. Memmove (move block of memory)
 - c. Strcpy (copy string)
 - d. Strncpy (copy characters from string)
2. Concatenation:
 - a. **Strcat (Concatenate strings)**
 - b. Strncat (Append character from string)
3. Comparison:
 - a. Memicmp (compare two blocks of memory)
 - b. **Strcmp (compare two string)**
 - c. Strcoll (compare two strings using locale)
 - d. Strncmp (compare characters of two strings)
 - e. Strxfrm (transform string using locale)
4. Searching:
 - a. Memchr (locate character in block of memory)
 - b. Strchr (locate first occurrence of character in string)
 - c. Strcspn (get span until character in string)
 - d. Strbrk (locate character in string)
 - e. Strrchr (locate last occurrence of character in string)
 - f. Strspn (get span of character set in string)
 - g. Strstr (locate substring)
 - h. Strtok (split string into tokens)
5. Other:
 - a. Memset (fill block of memory)
 - b. Strerror (get pointer to error message string)
 - c. **Strlen (get string length)**

Contoh penggunaan fungsi strcpy

```
#include <stdio.h>

int main()
{
    char a[] = "Halo";
    char b[10];

    strcpy(b,a);
    printf("%s\n", b);
    return 0;
}
```



```
[Running] cd "c:\Users\A"
Halo
[Done] exited with code=
```




MAGISTER TEKNIK INFORMATIKA

MATRIKULASI – STRUKTUR DATA

Contoh fungsi strcat

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[] = "Halo";
    char b[] = "kawan";
    char c[20];

    strcpy(c,a);
    strcat(c,b);
    printf("%s\n", c);
    return 0;
}
```

```
[Running] cd "c:\Users\Asus\Desktop\tugas\St
Halokawan
```

```
[Done] exited with code=0 in 0.914 seconds
```

Contoh fungsi strcmp

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[] = "Halo";
    char b[] = "Hai";
    char c[] = "Halo";

    if (strcmp(a,b) == 0)
    {
        printf("string a sama dengan b\n");
    }
    else
    {
        printf("string a tidak sama dengan b\n");
    }

    if (strcmp(a,c) == 0)
    {
        printf("string a sama dengan c\n");
    }
    else
    {
        printf("string a tidak sama dengan c\n");
    }

    return 0;
}
```

```
[Running] cd "c:\Users\Asus\Desk
string a tidak sama dengan b
string a sama dengan c
```

```
[Done] exited with code=0 in 0.9
```

Contoh Fungsi strlen

```
#include <stdio.h>
#include <string.h>
int main()
{
    char a[] = "halo";
    printf("panjang string a adalah %d\n", strlen(a));
    return 0;
}
```

```
[Running] cd "c:\Users\Asus\D
panjang string a adalah 4
```

```
[Done] exited with code=0 in
```