

Editorial Si Paling Ribet

Dasar Pemrograman G 2022

Algoritma/Pengetahuan yang Dibutuhkan

Modul 4

Pendekatan/Observasi

Dengan menggunakan cara pembagian kasus untuk inisialisasi variabel biasa dengan menggunakan pointer jika menggunakan variabel pointer maka '&' dihapus dan dicocokkan untuk variabel jika cocok dimasukkan ke unsur struct tersebut. Untuk kasus modify dibagi menjadi 3 kasus lagi yaitu dimana modify mengganti suatu variabel dengan angka maka cocokkan variabel pada struct jika cocok unsur nilai pada struct diganti dengan angka yang baru, kasus selanjutnya menukar nilai suatu variabel dengan variabel lain, praktikan dapat dapat mencocokkan alamat dan variabel pointer alamat jika keduanya sudah cocok pada suatu unsur struct maka nilai dari struct tersebut diganti dengan nilai yang baru, kasus terakhir merubah suatu variabel dengan menggunakan variabel pointer, praktikan dapat mencocokkan variabel pointer dari struct jika ditemukan kesamaan maka ganti nilai dengan nilai yang baru. Untuk perintah print lakukan pencocokan variabel jika cocok maka print nilai dari suatu variabel tersebut.

Source Code

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <math.h>
typedef unsigned long long llu;
typedef long long ll;
struct lelah
{
    char variabel[100];
    char variabel_alamat[100];
    char alamatnya_ke_variabel[100];
    int nilainya;
};

int main()
{
    int N, x = 0;
    int lawas, baru;
    scanf("%d", &N);
    struct lelah input[N];
    char arr[100];
    for (int i = 0; i < N; i++)
    {
        scanf("%s", arr);
        if (strcmp(arr, "init") == 0)
        {
            char arr1[100];
```

```

scanf("%s", arr1);
if (strcmp(arr1, "*") == 0) //! belum
{
    char nn[100], mm[100];
    scanf("%s %s", nn, mm);
    int l_mm = strlen(mm);
    char new[100];
    for (int j = 0; j < l_mm; j++)
    {
        new[j] = mm[j + 1];
    }
    for (int i = 0; i < x; i++)
    {
        if (strcmp(input[i].variabel, new) == 0)
        {
            strcpy(input[i].variabel_alamat, nn);
            strcpy(input[i].alamatnya_ke_variabel, mm);
        }
    }
}
else
{
    strcpy(input[x].variabel, arr1);
    scanf("%d", &input[x].nilainya);
    x++;
}
}
else if (strcmp(arr, "modify") == 0)
{
    char hh[100];
    scanf("%s", hh);
    if (strcmp(hh, "*") == 0) //! belum
    {
        char ll[100];
        int ww;
        scanf("%s %d", ll, &ww);
        for (int j = 0; j < x; j++)
        {
            if (strcmp(input[j].variabel_alamat, ll) == 0)
            {
                input[j].nilainya = ww;
            }
        }
    }
    else
    {
        char aye2[100];
        scanf("%s", aye2);
    }
}

```

```

        if (aye2[0] == '&') //! belum
        {
            // int lawas, baru;
            for (int i = 0; i < x; i++)
            {
                if (strcmp(input[i].variabel_alamat, hh) == 0)
                {
                    lawas = i;
                    break;
                }
            }
            for (int j = 0; j < x; j++)
            {
                if (strcmp(input[j].alamatnya_ke_variabel, aye2) ==
0)
                {
                    baru = j;
                    break;
                }
            }
            input[lawas].nilainya = input[baru].nilainya;
        }
        else
        {
            int kaka = atoi(aye2);
            for (int j = 0; j < x; j++)
            {
                if (strcmp(input[j].variabel, hh) == 0)
                {
                    input[j].nilainya = kaka;
                }
            }
        }
    }
}
else if (strcmp(arr, "print") == 0)
{
    char arr2[100];
    scanf("%s", arr2);
    for (int j = 0; j < x; j++)
    {
        if (strcmp(input[j].variabel, arr2) == 0)
        {
            printf("%d\n", input[j].nilainya);
        }
    }
}
}
}

```

}