LAPORAN PRAKTIKUM

POINTER (DYNAMIC ARRAY)

1031101 – Dasar Programming

Disusun Oleh:

Nama : Putri Geraldine Alexsandra Sihombing

NIM : 11323009

Diploma III Teknologi Informasi IT DEL 2023

Situoluama, Laguboti 2023

2023/2024

A. Praktikum

1. Program 1: pointer dan elemen array

Array disimpan dalam memori yang berurutan, array indeks terendah adalah elemen pertama array.

Berikut kode dalam pointer dan elemen array:

D: > KULIAH > SEMESTER 1 > Daspro > Week 11 > Prak_Daspro_W11S02_11323009 > C pointer_n_array.c

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include <stdio.h>

int array[5] = {100, 200, 300, 400, 500};
int main()
{
    int *pointer;

    printf ("Alamat elemen ke-1 array (&array [0])  = %d\n", &array[0]);
    printf ("Alamat elemen ke-1 array (array)       = %d\n\n", array);

    pointer= array;
    printf("Alamat elemen ke-1 array (pointer      = %d\n\n)", pointer);

    printf("Elemen pertama array    = %d\n", array[0]);
    printf("Elemen pertama array     = %d\n", *pointer);
}
```

Output dari hasil kode diatas:

```
D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>gcc pointer_n_array.c -o n

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>n
Alamat elemen ke-1 array (&array [0])  = 4210692
Alamat elemen ke-1 array (array)       = 4210692

Alamat elemen ke-1 array (pointer      = 4210692

)Elemen pertama array    = 100
Elemen pertama array     = 100

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>
```

2. Program 2: Menampilkan alamat dari elemen array

Alamat dari elemen array berbeda setiap variabelnya.

Berikut kode dari menampilkan Alamat dari elemen array:

D: > KULIAH > SEMESTER 1 > Daspro > Week 11 > Prak_Daspro_W11S02_11323009 > C alamat_elemen_array.c

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include <stdio.h>

int int_array[10], i;
float float_array[10];
double double_array[10];
   int main ()
{
    printf("\t\tInteger\t\tFloat\t\tDouble");
    printf("\n==================================");
    printf("============================");

    for (i=0; i<10; i++)
        printf("\n Element %d : \t%ld\t\t%ld\t\t%ld", i, &int_array[i], &float_array[i], &double_array[i]);
    printf("\n==================================");
    printf("============================\n");
    return 0;
}
```

Output dari code diatas yaitu:

```
D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>gcc alamat_elemen_array.c -o n

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>n
                Integer         Float           Double
==================================================================
 Element 0 :    4223264         4223200         4223104
 Element 1 :    4223268         4223204         4223112
 Element 2 :    4223272         4223208         4223120
 Element 3 :    4223276         4223212         4223128
 Element 4 :    4223280         4223216         4223136
 Element 5 :    4223284         4223220         4223144
 Element 6 :    4223288         4223224         4223152
 Element 7 :    4223292         4223228         4223160
 Element 8 :    4223296         4223232         4223168
 Element 9 :    4223300         4223236         4223176
==================================================================

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>
```

3. Program 3: Pointers used as arrays

   Pemanggilan data dengan memanggil adsress dari sebuah pointer secara berurutan.

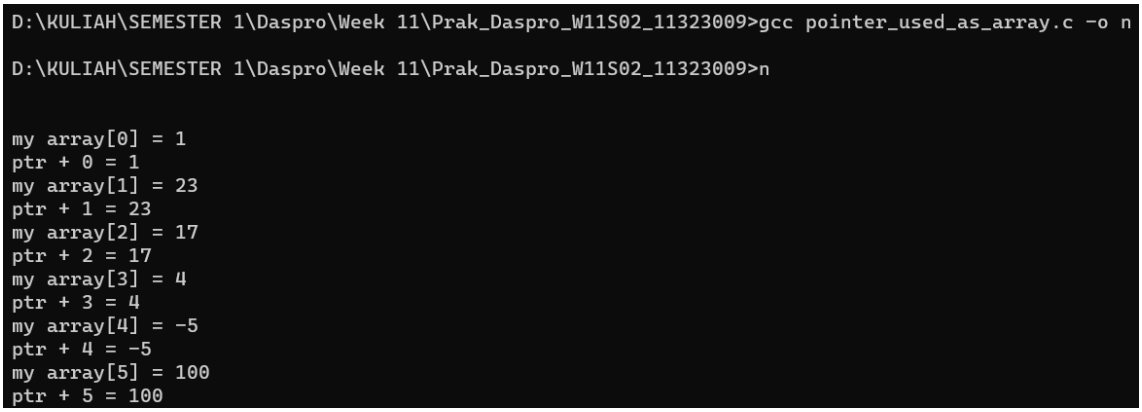
   Berikut kode dari pointers used as arrays:

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include <stdio.h>

int my_array[] = {1,23,17,4, -5, 100};
int *ptr;
int main(void)
{
    int i;
    ptr = &my_array[0];
    printf("\n\n");
    for (i=0; i<6; i++)
    {
        printf("my array[%d] = %d \n", i, my_array[i]);
        printf("ptr + %d = %d\n",i, *(ptr + i));
    }
    return 0;
}
```

   Output dari code diatas:



```
D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>gcc pointer_used_as_array.c -o n

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>n


my array[0] = 1
ptr + 0 = 1
my array[1] = 23
ptr + 1 = 23
my array[2] = 17
ptr + 2 = 17
my array[3] = 4
ptr + 3 = 4
my array[4] = -5
ptr + 4 = -5
my array[5] = 100
ptr + 5 = 100
```

4. Program 4: Dinamic Multidimensional Arrays

   Berikut kode dari dynamic multidimensional arrays:

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/
#include <stdio.h>
#include <stdlib.h>
int main ()
{
    int rows = 3;
    int cols = 5;
    int **arr = (int **)malloc(rows * sizeof(int *));

    for (int i =0; i<rows; i++){
        arr[i] = (int *)malloc(cols * sizeof(int));
    }

    int count = 1;
    for (int i = 0; i < rows; i++){
        for (int j = 0; j<cols; j++){
            arr[i][j] = count++;
        }
    }

    for(int i = 0; i < rows; i++){
        for(int j =0; j<cols; j++){
            printf("%d ", arr[i][j]);
        }
        printf("\n");
    }

    for (int i = 0; i < rows; i++){
        free(arr[i]);
    }
    free(arr);
    return 0;
}
```

Output dari kode diatas:

```
D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>gcc Dinamic_multidimensional_array.c -o n

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>n
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
```

5. Program 5: Comparison Pointer

   Berikut kode dari comparison pointer:

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/
#include <stdio.h>
#define MAX 10

int i_array[MAX] = {0,1,2,3,4,5,6,7,8,9};
int *i_ptr, count;
float f_array[MAX] = {.0, .1, .2, .3, .4, .5, .6, .7, .8, .9};
float *f_ptr;

int main (void)
{
    i_ptr = i_array;
    f_ptr = f_array;

    while(i_ptr <= &i_array[MAX-1]){
        printf("%d\t%f\n", *i_ptr++, *f_ptr++);
    }

    return 0;
}
```

Output dari kode diatas yaitu:

```
D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>gcc comparison_pointer.c -o n

D:\KULIAH\SEMESTER 1\Daspro\Week 11\Prak_Daspro_W11S02_11323009>n
0       0.000000
1       0.100000
2       0.200000
3       0.300000
4       0.400000
5       0.500000
6       0.600000
7       0.700000
8       0.800000
9       0.900000
```

B. Analisis

1. Gunakan kombinasi code berikut pada pointer_used_as_array.c

   a. Changes line B to: printf("ptr + %d = %d\n",i, *ptr++);

      Pada bagian ini output yang dihasilkan tidak memiliki perubahan apapun.

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include <stdio.h>

int my_array[] = {1,23,17,4, -5, 100};
int *ptr;
int main(void)
{
    int i;
    ptr = &my_array[0];
    printf("\n\n");
    for (i=0; i<6; i++)
    {
        printf("my array[%d] = %d \n", i, my_array[i]);
        printf("ptr + %d = %d\n",i, *ptr++);
    }
    return 0;
}
```

Output dari kode diatas:

```
my array[0] = 1
ptr + 0 = 1
my array[1] = 23
ptr + 1 = 23
my array[2] = 17
ptr + 2 = 17
my array[3] = 4
ptr + 3 = 4
my array[4] = -5
ptr + 4 = -5
my array[5] = 100
ptr + 5 = 100
```

b. Changes line B to:

printf("ptr + %d = %d\n",i, *(++ptr));

What happens?

Pada bagian ini terjadi perubahan output disebabkan oleh pemanggilan ++ptr telah berbeda

Kode:

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include <stdio.h>

int my_array[] = {1,23,17,4, -5, 100};
int *ptr;
int main(void)
{
    int i;
    ptr = &my_array[0];
    printf("\n\n");
    for (i=0; i<6; i++)
    {
        printf("my array[%d] = %d \n", i, my_array[i]);
        printf("ptr + %d = %d\n",i, *(++ptr));

    }
    return 0;
}
```

Output dari kode diatas:

```
my array[0] = 1
ptr + 0 = 23
my array[1] = 23
ptr + 1 = 17
my array[2] = 17
ptr + 2 = 4
my array[3] = 4
ptr + 3 = -5
my array[4] = -5
ptr + 4 = 100
my array[5] = 100
ptr + 5 = 2
```

c. In C &var_name[0] can be replaced with var_name

ptr=&my_array[0] is equivalent to ptr=my_array;

kode:

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include <stdio.h>

int my_array[0] ;
int *ptr;
int main(void)
{
    int i;
    ptr=&my_array[0];
    printf(" \n\n");

    for(int i=0; i<=5; i++){
        printf("myArray[%d]= %d\n\n", i, &my_array[i]);
    }
    return 0;
}
```

output :

```
myArray[0]= 4223092

myArray[1]= 4223096

myArray[2]= 4223100

myArray[3]= 4223104

myArray[4]= 4223108

myArray[5]= 4223112
```

2. Buatlah input untuk rows dan columns dari user pada Dinamic_multidimensional_array.c. Berikan hasilnya berupa screenshot dan hasil program

   Kode:

```c
/*
NIM : 11323009
*/
#include <stdio.h>
#include <stdlib.h>
int main (){
    int rows;
    int cols;
    printf("enter jumlah rows: ");
    scanf("%d", &rows);
    printf("enter jumlah cols: ");
    scanf("%d", &cols);

    int **arr = (int **)malloc(rows * sizeof(int *));

    for (int i = 0; i < rows; i++) {
        arr[i] = (int *)malloc(cols * sizeof(int));
    }
    int count = 1;
    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            arr[i][j] = count++;
        }
    }

    for (int i = 0; i < rows; i++) {
        for (int j = 0; j < cols; j++) {
            printf("%d ", arr[i][j]);
        }
        printf("\n");
    }

    for (int i = 0; i < rows; i++) {
        free(arr[i]);
    }
    free(arr);
    return 0;
```

   Output:

```
enter jumlah rows: 2
enter jumlah cols: 3
1 2 3
4 5 6
```

3. Diberikan array dengan element [5] = {2534, 4652, 8476, 2341, 4876}
    - Buat program yang dapat menampilkan Alamat memory dari setiap data yang disimpan.
    - Gambarkan setiap penyimpanan pada lokasi memory dari data yang disimpan

Kode:

```c
/*
Nama : Putri Geraldine Allexsandra Sihombing
NIM : 11323009
*/

#include<stdio.h>
#include<stdlib.h>

int main()
{
    int elemen[5] = {2534, 4652, 8476, 2341, 4876};

    printf("Elemen\t\t Alamat\n");
    printf("=======================\n");

    for (int j=0; j<5; j++)
    {
        printf("%d\t\t %d\n", elemen[j], &elemen[j]);
    }
}
```

Output:

```
Elemen          Alamat
=======================
2534            6422280
4652            6422284
8476            6422288
2341            6422292
4876            6422296
```