Praktikum 10 (2/5)

POINTER

A. TUJUAN

1. Menjelaskan tentang cara Mengakses Isi Suatu Variabel Melalui Pointer
2. Mengakses dan Mengubah isi Suatu Variabel Pointer

B. DASAR TEORI

Mengakses dan Mengubah isi Suatu Variabel Pointer

Contoh berikut memberikan gambaran tentang pengubahan isi suatu variabel secara tak langsung (yaitu melalui pointer). Mula-mula **pd** dideklarasikan sebagai pointer yang menunjuk ke suatu data bertipe *float* dan **d** sebagai variabel bertipe *float*. Selanjutnya

```
d = 54.5;
```

digunakan untuk mengisikan nilai 54,5 secara langsung ke variabel **d**. Adapun

```
pd = &d;
```

digunakan untuk memberikan alamat dari **d** ke **pd.** Dengan demikian **pd** menunjuk ke variabel **d.** Sedangkan pernyataan berikutnya

`*pd = *pd + 10;` (atau: `*pd += 10;`)

merupakan instruksi untuk mengubah nilai variabel **d** secara tak langsung. Perintah di atas berarti "jumlahkan yang ditunjuk ***pd*** dengan 10 kemudian berikan ke yang ditunjuk oleh **pd**", atau identik dengan pernyataan

```
d = d + 10;
```

Akan tetapi, seandainya tidak ada instruksi

```
pd = &d;
```

maka pernyataan

```
*pd = *pd + 10;
```

tidaklah sama dengan

```
d = d + 10;
```

C. PERCOBAAN

Untuk setiap program di bawah ini,

- gambarkan ilustrasi alokasi memori dari setiap baris pernyataan yang diproses
- perkirakan hasil eksekusinya

1.
```
#include <stdio.h>
main()
{
  float d = 54.5f, *pd;

  printf("Isi d mula-mula = %g\n", d);

  pd = &d;
  *pd += 10;

  printf("Isi d sekarang  = %g\n", d);
}
```

2.
```
#include <stdio.h>

main()
{
  int i1, i2, *p1, *p2;

  i1 = 9;
  p1 = &i1;
  i2 = *p1 / 2 - 2 * 3;
  p2 = p1;

  printf("i1=%d, i2=%d, *p1=%d, *p2=%d\n",i1,i2,*p1,*p2);
}
```

3.
```
#include <stdio.h>
main()
{
  int  count = 10, *temp, sum = 7;

  temp = &count;
  *temp = 32;
  temp = &sum;
  *temp = count;
  sum = *temp * 4;
  printf("count = %d, *temp = %d, sum = %d\n", count,
       *temp, sum );
}
```

4.
```
#include <stdio.h>
main()
{
  int  count = 13, sum = 9, *x, *y;

  x = &count;
  *x = 27;
  y = x;
  x = &sum;
  *x = count;
  sum = *x / 2 * 3;

  printf("count = %d, sum = %d, *x = %d, *y = %d\n",
     count,sum,*x,*y);
}
```

5.
```
#include <stdio.h>

int r, q = 7;

int go_crazy(int *, int *);

main()
{
  int *ptr1 = &q;
  int *ptr2 = &q;

  r = go_crazy(ptr1, ptr2);
  printf("q = %d, r = %d, *ptr1 = %d, *ptr2 = %d\n",
      q, r,*ptr1,*ptr2);

  ptr2 = &r;
```

```
go_crazy(ptr2, ptr1);
printf("q = %d, r = %d, *ptr1 = %d, *ptr2 = %d\n",
         q, r,*ptr1,*ptr2);
}

int go_crazy(int *p1, int *p2)
{
 int x = 5;

 r = 12;
 *p2 = *p1 * 2;
 p1 = &x;
 return *p1 * 3;
}
```

D. LAPORAN RESMI

Kumpulkan listing program, ilustrasi alokasi memorinya beserta hasil eksekusinya.