

Mata Kuliah : Dasar Dasar Pemrograman (Praktek)  
Kode Mata Kuliah : KBTI4104  
Waktu : Jumat (8.20 – 11.40)  
Jumlah SKS : 4 SKS  
Nama Dosen : Ani Rahmani  
Minggu ke : 12 (Dua Belas)  
Tanggal : 4-12-2015  
Judul Materi : Review UTS & Modular Programming

Hasil review UTS dari bu AN :

- Naming variabel, tolong diperhatikan, agar bisa di mengerti orang lain.
- Harus sesuai standar, gunakan header.
- Masih banyak salah logic, dan compile error.
- Ada juga yang hanya berisi Algoritma saja, tidak ada printf padahal online judge memeriksa output.
- Pemberian komentar.

Dalam Programming ada 9 peran variabel :

1. Tidak berisi curahan hati.
2. Tidak berlebihan, seperlunya saja.
3. Tidak menipu.
4. Tidak menuliskan komentar disetiap var/instruksi.
5. Tidak membuat komentar seperti di jurnal. Saya kurang paham maksudnya.
6. Tidak menuliskan komentar yang umum diketahui orang lain.
7. dan lain-lain.

Komentar ditulis ketika ada sesuatu yang belum terbaca disource code.

Ketika ada sebuah string dengan elemen 50, maka harus disiapkan 51 untuk "\0".

# Modular Programming

Kenapa harus modular?

1. Reusability
2. Mudah ditemukan kesalahannya.
3. Team work

Dalam modular di bagi menjadi dua bagian, yaitu :

1. Function  
Mempunyai 1 nilai kembalian.
2. Procedure  
Merubah state, Ada variabel yang nilai nya berubah.

Contoh modular programming dalam bahasa c

Tipe Module	Nama Program	Source Code
Procedure	Swapping Dua Variable	<pre>#include&lt;stdio.h&gt;  /*Prototype*/ void swap(int *a, int *b);  int main(){      /* Deklarasi */      int a,b;      /* Algoritma */      printf("\nMasukkan nilai a : ");      scanf("%d",&amp;a);      printf("\nMasukkan nilai b : ");</pre>

		<pre> scanf("%d",&amp;b);  // Cetak var a &amp; b sebelum di swap printf("Sebelum di swap\nVar a : %d\nVar b : %d\n",a,b);  // Memanggil modul prosedur swap swap(&amp;a,&amp;b);  // Cetak var a &amp; b sesudah di swap printf("Sesudah di swap\nVar a : %d\nVar b : %d\n",a,b);  return 0; }  void swap(int *a, int *b){     *a = *a+*b;     *b = *a-*b;     *a = *a-*b; } </pre>
Function	Perkalian	<pre> #include&lt;stdio.h&gt;  /*Prototype*/ int kali(int angka, int n); </pre>

		<pre>int main(){      /* Deklarasi */      int angka,n,hasil;      /* Algoritma */      printf("\nMasukkan angka : ");      scanf("%d",&amp;angka);      printf("\nKali : ");      scanf("%d",&amp;n);       // Memanggil modul fungsi kali dan di     // tampung ke dalam var hasil      hasil = kali(angka,n);      // Cetak hasil      printf("%d x %d = %d\n",angka,n,hasil);      return 0; }  int kali(int angka, int n){      int i,hasil;      for(i=1;i&lt;=n;i++){</pre>
--	--	--

		<pre>        hasil += angka;      }      return hasil;  }</pre>
--	--	---