

Pada semua program, tulis #include <stdio.h> sebelum int main() atau void main()

Semua program di-save dengan nama file berekstensi .cpp

CONTOH PROGRAM KECIL dalam BAHASA C untuk Praktikum MK DDP

boolean.h

```
#ifndef boolean_H
#define boolean_H
#define true 1
#define false 0
#define boolean unsigned char
#endif
```

```
-----
/* File           : hello.c                               */
/* Deskripsi      : Contoh menampilkan hello word        */
/* Dibuat       Oleh : Dosen DDP / 132 231 592           */
/*              Tgl  : 30/08/01                           */
void main()
{
    printf("hello\n");
}
```

```
-----
/* File           : asign.c                               */
/* Deskripsi      : Asign nilai integer dan print         */
/* Dibuat       Oleh : Dosen DDP / 132 231 592           */
/*              Tgl  : 30/08/01                           */

int main()
{
    /* Kamus */
    int i;

    /* program */
    printf("Hello \n");
    i=5;
    printf("Ini nilai i : %d \n ",i);
    return 0;
}
```

```
-----
/* File           : asgd11.c                               */
/* Deskripsi      : Assignment float dan print            */
/* Dibuat       Oleh : Dosen DDP / 132 231 592           */
/*              Tgl  : 30/08/01                           */

int main()
{

    /* Kamus */
    float f;
    long double f11;

    /* Program */
    f=20.0f;
    f11=10.0L;

    /* Program untuk menampilkan Hasil assign ke layar */
    printf("Nilai f      = %f \n",f);
    printf("Nilai f11   = %ld \n",f11);
    return 0;
}
```

```

/* File           : asign1.c                               */
/* Deskripsi      : Contoh assignment dan print            */
/* Dibuat       Oleh : Dosen DDP / 132 231 592            */
/*              Tgl  : 30/08/01                           */

#include <limits.h>

int main()
{
    /* Kamus */

    int i;
    long int ii;

    /* Program */

    printf("Hello \n");
    i= 1234;
    ii=123456;

    printf("Ini nilai i=1234  = %d \n ",i);
    printf("Ini nilai ii=123456 = %ld\n",ii);

    /* Print nilai batas integer */
    printf("Min dan Max integer = %d , %d \n",INT_MIN,INT_MAX);
    printf("Max long integer      = %ld \n",LONG_MAX);
    return 0;
}

-----
/* File           : asign2.c                               */
/* Deskripsi      : Berisi contoh sederhana untuk         */
/*              mendefinisikan variabel-variabel          */
/*              bil bulat short int, int, long int        */
/* Dibuat       Oleh : Dosen DDP / 132 231 592            */
/*              Tgl  : 30/08/01                           */

int main()
{
    /* Kamus */
    short ks = 1;
    int ki=1;
    long kl=10000;
    char c=65;          /*inisialisasi karakter dgn integer*/
    char cl='Z';         /*inisialisasi karakter dgn karakter*/
    float x=1.55;

    /* Program */

    /* penulisan karakter sebagai karakter */
    printf("Karakter = %c \n",c);
    printf("Karakter = %c \n",cl);

    /* Penulisan karakter sebagai integer */
    printf("Karakter = %d \n",c);
    printf("Karakter = %d \n",cl);

    printf("Bilangan integer (short) = %d \n",ks);
    printf("\t\t(int)   = %d \n",ki);
    printf("\t\t(long)  = %ld\n",kl);
    printf("Bilangan real = %8.3f \n",x);

    return 0;
}

```

```

-----
/* File      : incr.c                               */
/* Deskripsi  : Efek dari operator                   */
/* Dibuat    Oleh : Dosen DDP / 132 231 592         */
/*          Tgl  : 30/08/01                          */
int main()
{
/* kamus */
    int i,j;

/* Algoritma */

    i = 3;
    j = i++;
    printf("Nilai i : %d\nNilai j : %d\n", ++i, j);
    return 0;
}

-----
/* File      : oper1.c                               */
/* Deskripsi  : Pemakaian beberapa operator terhadap bit */
/* Dibuat    Oleh : Dosen DDP / 132 231 592         */
/*          Tgl  : 30/08/01                          */

int main()
{

/* Kamus */
    int n=10;      /* 1010 */
    int x=1;       /* 0001 */
    int y=2;       /* 0010 */

/* Program */

    printf("n = %d \n",n);
    printf("x = %d \n",x);
    printf("y = %d \n",y);

    printf("n & 8  = %d \n",n&8);    /* 1010 And 1000 */
    printf("x & -8 = %d \n",x&-8);   /* 1 And 0111 */
    printf("y << 2 = %d \n",y<<2);  /* 10 ==> 1000 = 8 */
    printf("y >> 3 = %d \n",y>>3);  /* 10 ==> 0000 = 0 */

    return 0;
}
-----

```

```

/* File      : oprator.c */
/* Deskripsi  : Contoh pengoperasian variabel bertipe dasar */
/* Dibuat    Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 30-08-2001 */

int main()
#include "boolean.h"
{
/* Kamus */
    boolean Bool1, Bool2, TF ;
    int i, j, hsl;
    float x, y, res;

/* Algoritma */
    printf ("Utk program ini, baca teksnya dan tambahkan output \n");
    Bool1 = True; Bool2 = False;
    printf ("Bool1 = %d ", Bool1, " dan Bool2 = %d \n", Bool2);
    TF = Bool1 && Bool2; /* Operator And dalam boolean */
    printf("TF = Bool1 And Bool2 : %d \n", TF);
    TF = Bool1 || Bool2; /* Operator or dalam boolean */
    printf("TF = Bool1 Or Bool2 : %d \n", TF);
    TF = ! Bool1; /* Operator Not dalam boolean */
    printf("TF = Not Bool1 : %d \n", TF);
    TF = Bool1 ^Bool2; /* Operator XOR dalam boolean */
    printf("TF = Bool1 Xor Bool2 : %d \n", TF);

/* Operasi Numerik */
    i = 5; j = 2;
    printf ( "Nilai i = %d", i, " dan nilai j = %d\n", j);
    hsl = i+j;
    printf ("Hasil operasi i + j : adalah %d\n", hsl);
    hsl = i-j;
    printf ("Hasil operasi i - j : adalah %d\n", hsl);
    hsl = i / j; /* Dalam integer dikenal i/j */
    printf ("Hasil operasi i div j : adalah %d\n", hsl);
    hsl = i*j;
    printf ("Hasil operasi i * j : adalah %d\n ", hsl);
    hsl = i div j; /* pembagian bulat */
    printf ("Hasil operasi i div j : adalah %d\n", hsl);
    hsl = i mod j; /* Sisa pembagian bulat */
    printf ("Hasil operasi i mod j : adalah %d\n", hsl);

/* Operasi Numerik */
    x = 5.0; y = 2.0;
    printf ("Nilai x = %f", x, " dan nilai y = %f\n", y);
    res = x + y;
    printf ("Hasil operasi x + y : adalah %f\n", res);
    res = x - y;
    printf ("Hasil operasi x - y : adalah %f\n", res);
    res = x / y;
    printf ("Hasil operasi x / y : adalah %f\n", res);
    res = x * y;
    printf ("Hasil operasi x * y : adalah %f\n", res);

/* Operasi relasional numerik */
    printf ("Karena nilai i = %d", i, " dan nilai j = %d", j, " maka \n");
    TF = (i == j);
    printf ("Hasil Operasi TF = (i == j) adalah : %d\n", TF);
    TF = (i != j);
    printf ("Hasil Operasi TF = (i != j) adalah : %d\n", TF);
    TF = (i < j);
    printf ("Hasil Operasi TF = (i < j) adalah : %d\n", TF);
    TF = (i > j);
    printf ("Hasil Operasi TF = (i > j) adalah : %d\n", TF);
    TF = (i <= j);

```

```

printf ("Hasil Operasi TF = (i <= j) adalah : %d\n", TF);
    TF = (i >= j);
printf ("Hasil Operasi TF = (i >= j) adalah : %d\n", TF);

/* Operasi relasional numerik */
printf ("Karena nilai x = %f", x, " dan nilai y = %f", y, " maka \n");
TF = (x != y);
printf ("Hasil Operasi TF = (x != y) adalah : %d \n ", TF);
TF = (x < y);
printf ("Hasil Operasi TF = (x < y) adalah : %d \n", TF);
TF = (x > y);
printf ("Hasil Operasi TF = (x > y) adalah : %d \n", TF);
TF = (x <= y);
printf ("Hasil Operasi TF = (x <= y) adalah : %d \n ", TF);
TF = (x >= y);
printf ("Hasil Operasi TF = (x >= y) adalah : %d \n", TF);
    return 0;
}

-----
/* File          : oper2.c                                */
/* Deskripsi     : Pemakaian beberapa operator            */
/*              terhadap relational dan bit                */
/* Dibuat      Oleh : Dosen DDP / 132 231 592              */
/*              Tgl  : 31/08/01                            */

int main()
{

/* Kamus */
    char i,j;

/* Program */
    i=3; /* 00000011 biner */
    j=4; /* 00000100 biner */

    printf("i = %d \n",i);
    printf("j = %d \n",j);

    printf("i && j = %d \n",i && j);/* 1:op logik : true and true ==> true*/
    printf("i & j = %d \n",i & j); /* 0: 00000000 dalam biner */
    printf("i || j = %d \n",i||j); /* 1 opr logik, true or true ==>true */
    printf("i |j = %d \n",i|j); /* 7:00000111 biner */
    printf("i ^j = %d \n",i^j); /* 7:00000111 biner */
    printf("~i = %d \n",~i); /* -4:11111100 biner */

return 0;
}

-----

```

PERHATIKAN OPERATOR BOOLEAN DIBANDINGKAN OPERATOR BIT JIKA DIPAKAI DALAM EKSPRESI KONDISIONAL :

```
/* KAMUS */
char i, j;
i = 3; j = 4;
/* Program */
if (i && j) { ..... } /* true = true and true (LOGIKA) */
if (i & j) { ..... } /* false = 00000000 (BIT) */
if (i || j) { ..... } /* true = true and true (LOGIKA) */
if (i | j) { ..... } /* true = 00000111 (BIT) */
if (i ^ j) { ..... } /* true = 00000111 (BIT) */
if (-i) { ..... } /* true = 11111000 (BIT) */
```

```
-----
/* File      : exp.c                               */
/* Deskripsi  : Pemakaian operator kondisional     */
/* Dibuat    Oleh : Dosen DDP / 132 231 592        */
/*          Tgl  : 31/08/01                         */

int main()
{

/* Kamus */
    int x=1,y=2;

/* Program */

printf("x = %d \n",x);
printf("y = %d \n",y);
printf("Hasil ekspresi = (x<y) ? x:y = %d \n",(x<y)? x:y);

return 0;
}
```

```
-----
/* File      : expl.c                               */
/* Deskripsi  : Pembagian integer, casting          */
/* Dibuat    Oleh : Dosen DDP / 132 231 592        */
/*          Tgl  : 31/08/01                         */

int main()
{

/* Kamus */
    int x = 1;
    int y = 2;
    float fx;
    float fy;

/* Program */

printf("x/y (format integer) = %d \n",x/y);
printf("x/y (format float)   = %f \n",x/y);

/* supaya hasilnya tidak nol */
fx = x;
fy = y;
printf("fx/fy (format integer) = %d \n",fx/fy);
printf("fx/fy (format float)   = %f \n",fx/fy);

/* casting */
printf("float(x)/float(y) (format integer) = %d \n",float(x)/float(y));
printf("float(x)/float(y) (format float)   = %f \n",float(x)/float(y));
x = 10;
```

```

    y = 3;
    printf("x/y (format integer) = %d \n",x/y);
    printf("x/y (format float)    = %f \n",x/y);
    return 0;
}

```

```

-----
/* File           : oper3.c                               */
/* Deskripsi      : Operator terner                         */
/*               : ekspresi ditulis sebagai makro         */
/* Dibuat   Oleh : Dosen DDP / 132 231 592                */
/*               Tgl  : 31/08/01                           */

```

```

#define max(a,b) ((a>b) ? a: b)
int main()
{

/* Kamus */
    int i = 0; /* perhatikan int i, j = 0 bukan seperti ini */
    int j = 0;
    char c = 8;
    char d = 10;
    char e = max(c,d);
    int k = max(i,j);

/* Program */
    printf("Nilai e = %d \n",e);
    printf("Nilai k = %d \n",k);

    i = 2;
    j = 3;
    k = max(i++, j++);
    printf("Nilai k = %d \n",k);
    return 0;
}

```

```

-----
/* File           : baca.c */
/* Deskripsi      : contoh membaca integer kemudian menulis nilai yang dibaca */
/* Dibuat   Oleh : Dosen DDP / 132 231 592                */
/* Tanggal      : 30-08-2001 */

```

```

int main()
{
/* Kamus */
    int a;
/* Program */
    printf ("Contoh membaca dan menulis, ketik nilai integer : ");
    scanf ("%d", &a);
    printf ("Nilai yang dibaca : %d\n ", a);
    return 0;
}

```

```

-----
/* File           : bacakar.c */
/* Deskripsi      : contoh membaca char kemudian menulis nilai yang dibaca */
/* Dibuat   Oleh : Dosen DDP / 132 231 592                */
/* Tanggal      : 30-08-2001 */

```

```

int main()
{
/* Kamus */
    char cc;

```

```

/* Program */
printf("Hello\n");
printf ("Baca 1 karakter : ");
scanf ("%c", cc);
printf ("%c", cc);
printf ("Bye\n ");
return 0;
}

-----
/* File      : bacastr.c */
/* Deskripsi : deklarasi & alokasi string, kemudian mengisinya dgn membaca */
/* Dibuat   Oleh : Dosen DDP / 132 231 592      */
/* Tanggal  : 30-08-2001 */

int main()
{
/* Kamus */
char *str;
char *str1;
/* Program */
printf("\nBaca string, maks 20 karakter : ");
str=(char*) malloc (20* sizeof(char));
printf("Masukan sebuah string, maksimal 20 char : ");
scanf("%s",str);
printf("String yang dibaca : %s\n", str);
str1=(char*) malloc (20* sizeof(char));
strcpy(str1,str);
printf("String yang disalin : %s\n", str1);
return 0;
}

-----
/* File      : asgstr.c */
/* Deskripsi : deklarasi & alokasi string, kemudian mengisinya dgn membaca */
/* Dibuat   Oleh : Dosen DDP / 132 231 592      */
/* Tanggal  : 30-08-2001 */
int main()
{
/* Kamus */
char *str;
char *str1;
/* Program */
printf("\nBaca string, maks 20 karakter : ");
str=(char*) malloc (20* sizeof(char));
strcpy(str,"Ini string .... ");
printf("String yang diisikan : %s\n", str);
str1=(char*) malloc (20* sizeof(char));
strcpy(str1,str);
printf("String yang disalin : %s\n", str1);
return 0;
}

```


KONSTANTA DAN KALKULASI SEDERHANA

```
/* File      : konstant.c */
/* Deskripsi  : Membaca jari-jari, menghitung luas lingkaran */
/*           : Latihan pemakaian konstanta */
/* Dibuat    Oleh : Dosen DDP / 132 231 592      */
/* Tanggal   : 30-08-2001 */

int main()
{
    /* Kamus */
    const float pi = 3.1415;
    float r;

    /* program */
    /* baca data */
    printf("Jari-jari lingkaran = ");
    scanf("%f",&r);
    /* hitung dan tulis hasil */
    printf("Luas lingkaran = %f\n", pi*r*r);
    printf("Akhir Program \n");
    return 0;
}

-----

/* File      : konstan2.c */
/* Deskripsi  : Menghitung luas lingkaran, dari jari-jari yang dibaca */
/* Dibuat    Oleh : Dosen DDP / 132 231 592      */
/* Tanggal   : 30-08-2001 */

#define pi = 3.1415
int main()
{
    /* Kamus */
    float r, luas, kel;

    /* program */
    /* Baca data */
    printf("Jari-jari lingkaran = ");
    scanf("%f", &r);
    /* Hitung dengan rumus */
    luas = pi * r * r;
    /* Print hasil */
    printf("Luas lingkaran = %6.2f\n", luas);
    kel = 2 * pi * r;
    printf("Keliling lingkaran = %6.2f\n", kel);
    printf("Akhir Program \n");
    return 0;
}

-----
```

INSTRUKSI KONDISIONAL

```
/* File      : if1.c */
/* Deskripsi  : contoh pemakaian if satu kasus */
/*           : membaca nilai integer, menuliskan nilainya jika positif */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 6-09-2001 */
```

```
int main()
{
    /* Kamus */
    int a;

    /* Algoritma */
    printf("Contoh IF 1 kasus \n");
    printf("Ketikkan suatu nilai integer : ");
    scanf("%d",&a);
    if (a >= 0)
    {
        printf("Nilai a positif ...%d \n", a);
    }
    return 0;
}
```

```
/* File      : if2.c */
/* Deskripsi  : Contoh pemakaian if dua kasus komplementer */
/*           : membaca sebuah nilai integer */
/*           : menuliskan 'Nilai a positif, nilai a ', jika a >= 0 */
/*           : 'Nilai a negatif, nilai a ', jika a < 0 */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 6-09-2001 */
```

```
int main ()
{
    /* Kamus */
    int a;

    /* Algoritma */
    printf("Contoh IF dua kasus \n");
    printf("Ketikkan suatu nilai integer : ");
    scanf("%d",&a);
    if (a >= 0)
    {
        printf("Nilai a positif ...%d\n", a);
    } else /* a < = */
    {
        printf("Nilai a negatif ...%d\n", a);
    }
    return 0;
}
```

```

/* File      : if3.c */
/* Deskripsi  : Contoh pemakaian if dua kasus komplementer */
/*           : membaca sebuah nilai integer */
/*           : menuliskan "Nilai a positif, nilai a ", jika a > 0 */
/*           : "Nilai a samadengan nol, nilai a", jika a = 0 */
/*           : "Nilai a negatif, nilai a ", jika a < 0 */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal    : 6-09-2001 */

```

```

int main ()
{
/* Kamus */
    int a;

/* Algoritma */
    printf("Contoh IF tiga kasus \n");
    printf("Ketikkan suatu nilai integer : ");
    scanf("%d",&a);
    if (a > 0)
    {
        printf("Nilai a positif ...%d\n", a);
    } else
    if (a == 0)
    {
        printf("Nilai a samadengan nol %d\n", a);
    } else /* a < 0 */
    {
        printf("Nilai a negatif ...%d\n", a);
    }
    return 0;
}

```

```

/* File      : tempair.c */
/* Deskripsi  : contoh pemakaian IF tiga kasus : wujud air */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal    : 6-09-2001 */

```

```

int main()
{
/* Kamus */
    int T;

/* Algoritma */
    printf("Contoh IF tiga kasus \n");
    printf("Temperatur (dlm. derajat C) = ");
    scanf("%d",&T);
    if (T < 0)
    {
        printf("Wujud air beku %d \n", T);
    } else if ((T >= 0) && (T <= 100))
    {
        printf("Wujud air cair %d \n", T);
    } if (T > 100)
    {
        printf("Wujud air uap/gas %d\n",T);
    }
    return 0;
}

```

```

/* File      : kasus.c */
/* Deskripsi  : contoh kasus dengan instruksi switch */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/

int main()
{
/* Kamus */
    char cc;

/* Program */
    printf("Ketikkan sebuah huruf, akhiri dengan RETURN !\n ");
    scanf("%c",&cc);
    switch (cc)
    {
        case 'a' :{
            printf("Yang anda ketik adalah a\n");
            break; }
        case 'u' : {
            printf("Yang anda ketik adalah u\n");
            break; }
        case 'e' : {
            printf("Yang anda ketik adalah e\n");
            break; }
        case 'i' : {
            printf("Yang anda ketik adalah i\n");
            break; }
        default :
            printf("Yang anda ketik adalah huruf mati\n");
    }
    return 0;
}

```

```

-----
/* File      : Max2.c */
/* Deskripsi  : Maksimum dua bilangan yang dibaca */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/

int main()
{
/* Kamus */
    int a, b;
/* Algoritma */
    printf("Maksimum dua bilangan \n");
    printf("Ketikkan dua bilangan, pisahkan dengan RETURN : \n ");
    scanf("%d %d", &a, &b);
    printf("Ke dua bilangan : a = %d, b = %d \n ",a,b);
    if (a >= b)
    {
        printf("Nilai a yang maksimum %d \n", a);
    } else /* a < b */
    {
        printf("Nilai b yang maksimum %d \n", b);
    }
    return 0;
}

```

PENGULANGAN

```
/* File      : prifor.c */
/* Deskripsi  : Baca N , print 1 s/d N dengan FOR */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/
```

```
int main()
{
    /* Kamus */
    int i, N;
    /* Program */
    printf("Baca N, print 1 s/d N \n");
    printf("N = ");
    scanf("%d", &N);
    for (i = 1; i <= N; i++)
    {
        printf("%d \n",i);
    }
    printf("Akhir program \n");
    return 0;
}
```

```
-----
/* File      : prirep.c */
/* Deskripsi  : Baca N , print 1 s/d N dengan REPEAT */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/
```

```
int main()
{
    /* Kamus */
    int i, N;
    /* Program */
    printf("Nilai N > 0 = "); /* Inisialisasi */
    scanf("%d ",&N);
    i = 1; /* first elemen */
    printf("Print i dengan REPEAT : \n");
    do
    {
        printf("%d \n",i); /* Proses */
        i++; /* Next elemen */
    } while (i <= N); /* Kondisi pengulangan */
    printf("Akhir program \n");
    return 0;
}
```

```
-----
/* File      : prxrep.c */
/* Deskripsi  : REPEAT dengan pake penanganan kasus kosong */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/
```

```
int main()
{
    /* Kamus */
    int sum, x;

    /* Program */
    printf("Masukkan nilai x (int), akhiri dengan 999 = ");

    /* Inisialisasi */
    scanf("%d",&x);
    if (x == 999)
    {
        printf("Kasus Kosong \n"); }
}
```

```

else
{
    /* Minimal ada satu data yang dijumlahkan */
    sum = 0;          /* Inisialisasi; invarian !! */
    do
    {
        sum = sum + x;    /* proses */
        printf("Masukkan nilai x (int), akhiri dengan 999 : ");
        scanf("%d",&x);  /* Next elemen */
    } while (x != 999);   /* Kondisi pengulangan */
    printf("Hasil penjumlahan = %d \n", sum);    /* Terminasi */
}
return 0;
}

```

```

-----
/* File      : priter.c */
/* Deskripsi  : Baca N , print 1 s/d N dengan ITERATE */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/

```

```

int main()
{
    /* Kamus */
    int i,N;
    /* Program */
    printf("Nilai N >0 = ");    /* Inisialisasi */
    scanf("%d",&N);
    i=1;                        /* First elemen */
    printf("Print i dengan ITERATE : \n");
    for (;;)
    {
        printf("%d \n",i);      /* Proses */
        if (i==N)               /* Kondisi berhenti */
            break;
        else
        {
            i++;                /* Next elemen */
        }
    } /* i == N */
    printf("Akhir program \n");
    return 0;
}

```

```

-----
/* File      : prxiter.c */
/* Deskripsi  : Contoh baca nilai x, dan jumlahkan dengan ITERATE */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/

```

```

int main()
{
    /* Kamus */
    int sum, x;
    /* Program */
    printf("Masukkan nilai x (int), akhiri dengan 999 = ");
    /* Inisialisasi */
    scanf("%d",&x);
    if (x == 999)
    {
        printf("Kasus Kosong \n"); }
    else
    {
        /* Minimal ada satu data yang dijumlahkan */
        sum = x;          /* Inisialisasi; invarian !! */
        for (;;)
        {
            printf("Masukkan nilai x (int), akhiri dengan 999 : ");

```

```

        scanf("%d",&x);    /* Next elemen */
        if ( x == 999)
            break;
        else
            { sum = sum + x; }    /* proses */
    }
    printf("Hasil penjumlahan = %d \n", sum);    /* Terminasi */
}
return 0;
}

```

```

/* File      : priw.c*/
/* Deskripsi  : Baca N , print 1 s/d N dengan WHILE */
/* Dibuat Oleh : Dosen DDP / 132 231 592    */
/* Tanggal   : 06-09-2001*/

int main()
{
    /* Kamus */
    int i, N;
    /* Program */
    printf("Nilai N >0  = ");    /* Inisialisasi */
    scanf("%d", &N);
    i=1;    /* First elemen */
    printf("Print i dengan WHILE : \n");
    while (i <= N)    /* Kondisi pengulangan */
    {
        printf("%d \n",i);    /* Proses */
        i++;    /* Next elemen */
    } /* ( i > N ) */
    printf("Akhir program \n");
    return 0;
}

```

```

/* File      : priwhile1.c */
/* Deskripsi  : Baca N , print 1 s/d N dengan WHILE (ringkas) */
/* Dibuat Oleh : Dosen DDP / 132 231 592    */
/* Tanggal   : 06-09-2001*/

int main()
{
    /* Kamus */
    int N;
    int i = 1;
    /* Program */
    printf("Nilai N >0  = ");    /* Inisialisasi */
    scanf("%d", &N);
    printf("Print i dengan WHILE (ringkas) : \n");
    while (i <= N)    /* Kondisi pengulangan */
    {
        printf("%d \n",i);    /* Proses */
        i++;    /* Next elemen */
    } /* ( i > N ) */
    printf("Akhir program \n");
    return 0;
}

```

```

/* File      : prxwhile.c*/
/* Deskripsi  : Contoh baca nilai x, Jumlah nilai yang dibaca dgn WHILE */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001*/

int main()
{
/* Kamus */
    int sum, x;
/* Program */
    sum =0;                                /* Inisialisasi */
    printf("Masukkan nilai x (int), akhiri dengan 999 = ");
    scanf("%d", &x);
    while (x != 999)                        /* Kondisi pengulangan */
    {
        sum = sum + x;
        printf("Masukkan nilai x (int), akhiri dengan 999 = ");
        /* Proses */
        scanf("%d",&x);                    /* Next elemen */
    }
    printf("Hasil penjumlahan = %d \n",sum);
    return 0;
}

-----
/* File      : forefer.c */
/* Deskripsi  : Loop terus menerus */
/* Dibuat Oleh : Dosen DDP / 132 231 592 */
/* Tanggal   : 06-09-2001 */

int main ()
{
    /* kamus */

    #define true 1

    /* program */

    printf("Program akan looping, akhiri dengan ^c ");
    while (true)
    {
        printf("Print satu baris .... \n");
    }
    return 0;
}

```


ARRAY

```
/* File      : tabel.c */
/* Deskripsi  : latihan array statis : mengisi dgn assignment, menulis */
/* Dibuat    Oleh : Dosen DDP / 132 231 592      */
/* Tanggal   : 07-09-2001 */
int main()
{
    /* kamus */
    int i;
    int tab[10];          /* Cara mengacu elemen ke-i : tab[i] */
    int N;
    /* program */
    N = 5;
    printf("Isi dan print tabel untuk indeks 1..5 \n");

    /* Isi dengan assignment */
    for (i = 1; i <= N; i++) {
        tab[i] = i;
    }

    /* Traversal ; print */
    for (i = 1; i <= N; i++) {
        printf("i = %d  tab[%d] = %d \n", i, i, tab[i]);
    }
    return 0;
}

-----
/* File      : tabel4.c */
/* Deskripsi  : tujuan sama dengan tabel.c,          */
/*           : hanya pada script ini digunakan struktur while */
/*           : dan ada penyederhanaan proses      */

/* Dibuat Oleh : Dosen DDP / 132 231 591      */
/* Tanggal   : 07-09-2001 */

#include <stdio.h>
int main()
{
    /* kamus */
    int i;
    int tab[10];          /* Cara mengacu elemen ke-i : tab[i] */
    int N;
    /* program */
    N = 5;
    printf("Isi dan print tabel untuk indeks 1..5 \n");

    /* Penyederhanaan proses dari proses yang ditulis pada program tabel.c */
    i = 1;
    while ( i <= N )
    {
        tab[i] = i;
        /* Traversal ; print */
        printf("i = %d  tab[%d] = %d \n", i, i, tab[i]);
        i++;
    }

    return 0;
}
```