

MODUL PRAKTIKUM STRUKTUR DATA JURUSAN TEKNIK INFORMATIKA



**Disusun oleh
Bangun Wijayanto, S.T.,M.Cs.**

Modul praktikum ini milik
Laboratorium Jurusan Teknik Informatika
Universitas Jendral Soedirman
Dilarang untuk menggandakan tanpa seijin pihak Jurusan Teknik Informatika
Modul praktikum dikembalikan setelah selesai praktikum

**Laboratorium Teknik Informatika
Universitas Jendral Soedirman
Gedung D Lantai 3
Jl. Mayjen Sungkono Km.5 Blater Purbalingga
Telp/Fax :(0281) 6596700**

Mengenai C

Sebagai suatu bahasa pemrograman, C seperti halnya pascal maupun fortran. Nilai disimpan dalam suatu variable. Program terstruktur dan dedefinisikan serta melakukan pemanggilan fungsi.

Bahasa C merupakan bahasa yang sukses. Mulai dikembangkan oleh Brian Kernighan dan Dennis Ritchie pada awal tahun '70-an, bahasa C banyak mengalami berbagai kemajuan, perubahan, dan hingga kini (30 tahun lebih!) bahasa C masih banyak digunakan dan dipelajari. Belajar menjadi programmer professional seakan belum lengkap tanpa mempelajari C. Tidak mengherankan bila beberapa universitas menjadikan bahasa C sebagai bahasa pemrograman yang wajib. Sistem-sistem operasi yang terkemuka seperti UNIX dan Windows juga dibangun dengan menggunakan bahasa C (dan juga C++).

Sistem operasi berbasis UNIX dituliskan dalam bahasa C. Dalam kenyataannya C di buat untuk diimplementasikan secara spesifik dalam system operasi UNIX .

Tujuan Praktikum

Tujuan praktikum ini adalah mengenalkan mahasiswa pada bahasa pemrograman C. Setelah mempelajari algoritma pemrograman diharapkan mahasiswa dapat menerapkannya pada sejumlah bahasa pemrograman tanpa terpaku pada salah satu bahasa pemrograman.

Tata Tertib Praktikum

Praktikum dilaksanakan sebagai salah satu point dalam penilaian nilai akhir sehingga dituntut kedisiplinan didalam pelaksanaan praktikum, sehingga dalam praktikum pemrograman ini dibuat beberapa aturan sebagai berikut:

1. Praktikan wajib datang tepat waktu
2. 10 menit sebelum praktikum akan diadakan evaluasi oleh asisten
3. Praktikan wajib membawa kartu absensi praktikum

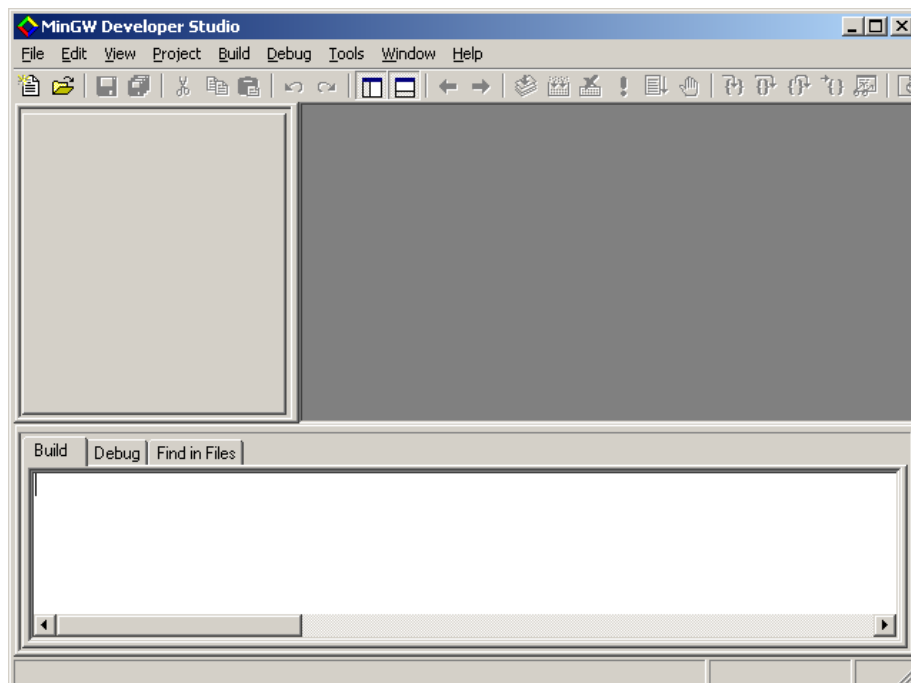
4. Seluruh mahasiswa peserta praktikum wajib membuat laporan praktikum, baik yang mengambil matakuliah baru maupun mengulang
5. **Tidak diadakan inhall bagi mahasiswa**, kecuali dengan alasan-alasan mendesak yang dapat diterima dengan persetujuan dosen pengampu.
6. Ketertiban dan keaktifan saat mengikuti praktikum dijadikan salah satu point penilaian oleh asisten.
7. Tujukan jalannya program anda pada asisten.
8. Modul di kumpulkan kembali setelah praktikum.

Perkakas dan kompiler

Sebagai penghargaan terhadap hak kekayaan intelektual dalam hal ini penggunaan perangkat lunak, maka seluruh tools (kakas) dan kompiler dalam praktikum menggunakan perangkat lunak bebas berlisensikan GNU.

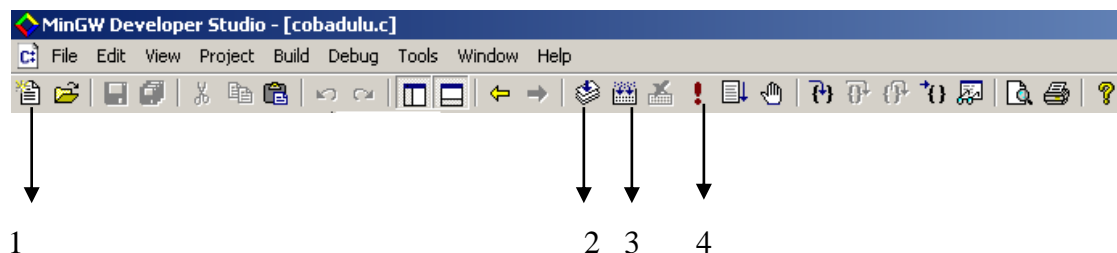
Mengenal IDE

Dalam praktikum ini akan digunakan tools (kakas) MinGW Developer Studio. MinGW Developer Studio adalah IDE (*Integrated Development Environment*/Lingkungan pengembangan) yang digunakan sebagai editor. MinGW Developer Studio melakukan kompilasi terhadap kode sumber bahasa C yang anda buat dengan GNU GCC (<http://gcc.gnu.org/>). GCC adalah GNU Compiler Collection, yang berisi front end C, C++, Objective-C, Fortran, Java, dan Ada



Gambar 1. MinGW Developer Studio

Untuk menjalankan MinGW Developer Studio klik pada icon yang terdapat di desktop atau pada start → MinGW Developer Studio

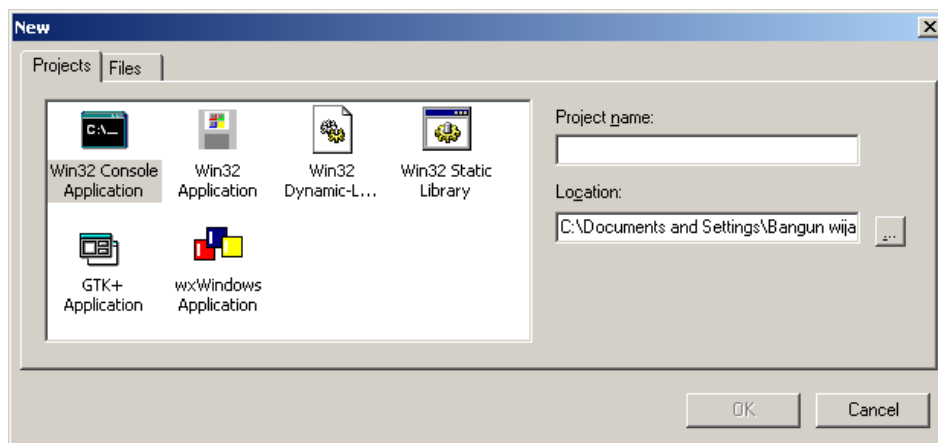


1. Tombol untuk membuat file C baru
2. Tombol untuk melakukan kompilasi program
3. Tombol untuk membangun aplikasi (build)
4. Tombol untuk menjalankan aplikasi

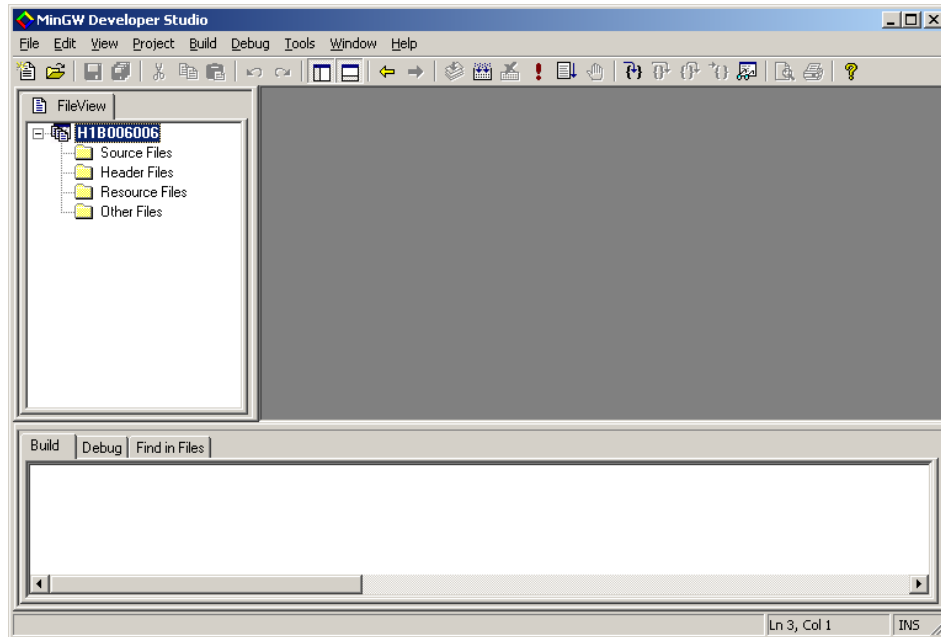
Membuat Project Baru

Dalam praktikum ini usahakan anda menggunakan 1 project saja. Berikut adalah langkah langkah membuat project baru

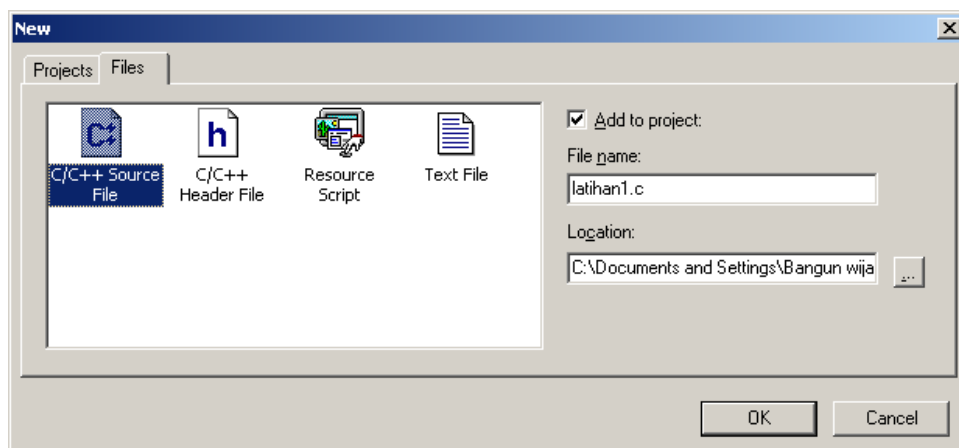
1. Buka MinGW Developer Studio
2. klik file new akan muncul tampilan sebagai berikut



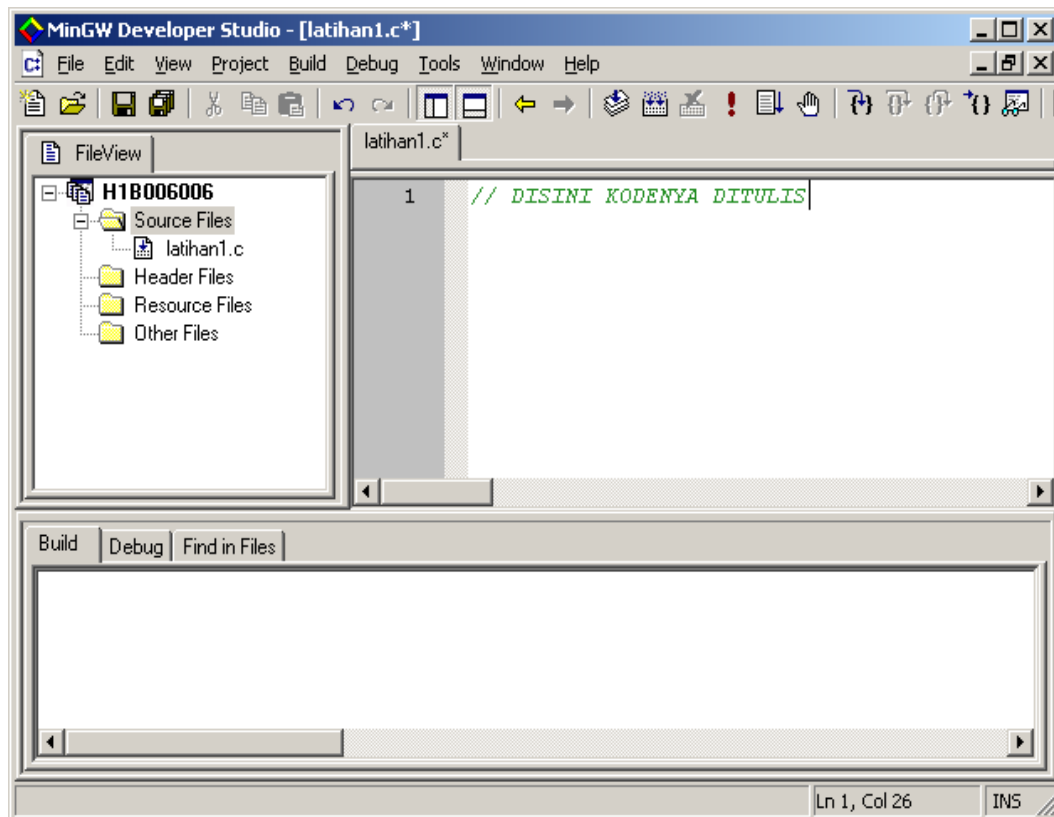
3. Pilih Win32 Console Application
4. Tuliskan nama project anda dengan NIM anda misal H1B006006 dan lokasi penyimpanan project anda
5. klik OK, maka akan muncul tampilan sebagai berikut



6. Untuk mengerjakan praktikum klik tombol nomor 1 (lihat gambar petunjuk tombol) akan muncul tampilan sebagai berikut



7. Pilih C/C++ Source file, pada bagian file name isikan nama file yang akan dibuat misal latihan1.c kemudian tekan OK
8. sekarang anda telah membuat file latihan1.c file tersebut bisa anda edit



9. Pada FileView terlihat bahwa project H1B006006 memiliki satu file yakni latihan1.c tuliskan kodenya pada bagian editor yang ada disebelah kanan
10. untuk melakukan kompilasi dan pembuatan serta pengujian program silahkan gunakan tombol yang telah dijelaskan diatas.

Hasil Aplikasi

Hasil aplkasi yang telah terkompilasi bisa di copy atau di ambil pada folder

<folder penyimpanan project>\\Debug\\

misal

C:\\Documents and Settings\\Bangun wijayanto\\My Documents\\coba\\Debug

Unit 1

Pointer & Linked List

1. IsiPointer.c

Buatlah file IsiPointer.c dan ketikkan kode berikut

```
#include<stdio.h>
int A,*B
void main()
{
    A=10;
    B=&A;
    *B=20;
    printf("%d",A);
}
```

a. Manakah yang dimaksud sebagai variabel pointer

Jawaban anda:

b. Tuliskan Output program tersebut

Jawaban anda:

c. Fungsi Pointer adalah

Jawaban anda:

1. LinkedList.c

Buatlah file LinkedList.c dan ketikan kode berikut

```
#include<stdio.h>
#include<stdlib.h>
#define EMPTY_LIST  NULL

typedef struct tnode *address;

struct tnode{
    int data;
    address next;
}node;

address first;
address last;
address current;
address alokasikan(int data){
    address alamat_alokasibaru;
    alamat_alokasibaru=(address)malloc(sizeof(node));
    alamat_alokasibaru->data=data;
    alamat_alokasibaru->next= NULL;
    return alamat_alokasibaru;
}
void insertfirst(address p){
    p->next=first;
    first=p;
}
void cetak(address p){
    do{
        printf("datanya %d\n",p->data);
        p=p->next;
    }while(p !=NULL);
}

main(){
    first=NULL;
    current=alokasikan(10);
    insertfirst(current);
    current=alokasikan(12);
    insertfirst(current);
    current=alokasikan(15);
    cetak(first);
}
```

a. Jelaskan apa yang dimaksud dengan linked list

Jawaban anda:

b. Sebutkan variabel pointer yang terdapat pada program diatas beserta fungsinya

Jawaban anda:

c. Tuliskan luaran program diatas

Jawaban anda:

d. Jelaskan fungsi insertfirst pada program diatas

Jawaban anda:

e. Ilustrasikan proses yang terjadi pada bagian main dengan menggunakan gambar

Jawaban anda:

Program akan dilengkapi sehingga bagian main program menjadi sebagai berikut

```
main() {  
    first=NULL;  
    current=alokasikan(10);  
    insertfirst(current);  
    current=alokasikan(12);  
    insertfirst(current);  
    current=alokasikan(15);  
    insertLast(current);  
    cetak(first);  
    deletefirst();  
    printf("sesudah dihapus satu\n");  
    cetak(first);  
}
```

f. Buatlah method insertLast

Jawaban anda:

```
void insertLast(address p) {
```

g. Buatlah method insertLast

Jawaban anda:

```
void DeleteFirst() {
```

- h. Ilustrasikan proses yang terjadi pada bagian main dengan menggunakan gambar

Jawaban anda:

Telah diperiksa	Tanggal :	Modul 1
Dengan ini praktikan telah menyetujui dan menyatakan bahwa aplikasi telah berjalan serta selesainya pengisian jawaban dari tiap pertanyaan yang ada		
Assisten	Praktikan	Nilai

Unit 2

Stack

1. Stack.c

Buatlah file Stack.c dan ketikan kode berikut

```
#include<stdio.h>
#include<stdlib.h>
#define MAXSTACK 2

typedef struct{
int item[MAXSTACK];
int count;
}Stack;
Stack tumpukan;

int isFull(Stack *x){
    int hasil=0;
    if ((x->count)>=MAXSTACK)
        hasil=1;
    return hasil;
}

int isEmpty(Stack *x){
    int hasil=0;
    if ((x->count)==0){
        hasil=1;
    }
    return hasil;
}

void membuatStack(Stack *x){
    x->count=0;
}

void push(int data,Stack *x){
    if (isFull(x)==1 ){
        printf("Stack Penuh\n");
    }else{
        x->item[x->count]=data;
        ++(x->count);
        printf("berhasil memasukan\n");
    }
}

void cetak(){
    for (i=0;i<MAXSTACK;i++){
        printf("isi index %d %d \n", i, tumpukan.item[i]);
    }
}
```

...lanjutan

```
main() {  
    int i;  
    membuatStack(&tumpukan);  
    printf("masukan 3\n");  
    push(3, &tumpukan);  
    printf("masukan 5\n");  
    push(5, &tumpukan);  
    printf("masukan 6\n");  
    push(6, &tumpukan);  
    printf("masukan 7\n");  
    push(7, &tumpukan);  
    cetak();  
}
```

- a. Jelaskan apa yang dimaksud dengan Stack

Jawaban anda:

- b. Tuliskan Luaran dari program diatas

Jawaban anda:

- c. Jelaskan dengan detail proses yang terjadi pada method push

Jawaban anda:

- d. Ilustrasikan dengan gambar proses yang terjadi pada bagian main program diatas

Jawaban anda:

Diinginkan pada bagian main program diatas menjadi seperti berikut

```
main() {  
    int i;  
    membuatStack(&tumpukan);  
    printf("masukan 3\n");  
    push(3,&tumpukan);  
    printf("masukan 5\n");  
    push(5,&tumpukan);  
    printf("masukan 6\n");  
    push(6,&tumpukan);  
    printf("masukan 7\n");  
    push(7,&tumpukan);  
    cetak();  
    pop(&tumpukan);  
    pop(&tumpukan);  
    pop(&tumpukan);  
    pop(&tumpukan);  
    cetak();  
}
```

e. Buatlah method pop

Jawaban anda:

```
void pop(Stack *x){
```


- f. Jelaskan dengan detail proses yang terjadi pada method pop

Jawaban anda:

- g. Ilustrasikan dengan gambar proses yang terjadi pada bagian main program diatas

Jawaban anda:

Unit 3

Queue

```
#include<stdio.h>
#include<stdlib.h>
#define MAXQUEUE 2
typedef struct{
    int count;
    int front;
    int rear;
    int item[MAXQUEUE];
}QUEUE;
QUEUE antrian;
void inisialisasi(QUEUE *Q){
    Q->count=0;
    Q->front=0;
    Q->rear=0;
}
int isEmpty(QUEUE *Q){
    int hasil=0;
    if (Q->count==0){
        hasil=1;
    }
    return hasil;
}

int isFull(QUEUE *Q){
    int hasil=0;
    if (Q->count==MAXQUEUE){
        hasil=1;
    }
    return hasil;
}
void insert(QUEUE *Q, int data){
    if(isFull(&antrian)==1){
        printf("maaf sudah full");
    }else{
        Q->item[Q->rear]=data;
        Q->rear=(Q->rear+1)%MAXQUEUE;
        ++(Q->count);
    }
}
}
main(){
    inisialisasi(&antrian);
    printf("masukan 3\n");
    insert(&antrian,3);
    printf("masukan 6\n");
    insert(&antrian,6);
}
```

a. Jelaskan apa yang dimaksud dengan Queue

Jawaban anda:

b. Sebutkan variabel pointer yang terdapat pada program diatas beserta fungsinya

Jawaban anda:

c. Tuliskan luaran program diatas (anda harus membuat fungsi print terlebih dahulu

Jawaban anda:

d. Jelaskan fungsi insert pada program diatas

Jawaban anda:

e. Ilustrasikan proses yang terjadi pada bagian main dengan menggunakan gambar

Jawaban anda:

UNIT 4

Searching & Sorting

Sorting adalah proses menyusun elemen – elemen dengan tata urut tertentu dan proses tersebut terimplementasi dalam bermacam aplikasi. Kita ambil contoh pada aplikasi perbankan. Aplikasi tersebut mampu menampilkan daftar *account* yang aktif. Hampir seluruh pengguna pada sistem akan memilih tampilan daftar berurutan secara *ascending* demi kenyamanan dalam penelusuran data. Beberapa macam algoritma *sorting* telah dibuat karena proses tersebut sangat mendasar dan sering digunakan. Oleh karena itu, pemahaman atas algoritma –algoritma yang ada sangatlah berguna.

Binary search

4.1. Buatlah Algoritma berikut pada file BinarySearch.c lalu lakukan pengujian

```
#include <stdio.h>

#define TRUE 0
#define FALSE 1

int main(void) {
    int array[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    int left = 0;
    int right = 10;
    int middle = 0;
    int number = 0;
    int bsearch = FALSE;
    int i = 0;

    printf("ARRAY: ");
    for(i = 1; i <= 10; i++)
        printf("[%d] ", i);

    printf("\nNomor yang akan dicari: ");
    scanf("%d", &number);

    while(bsearch == FALSE && left <= right) {
        middle = (left + right) / 2;

        if(number == array[middle]) {
            bsearch = TRUE;
            printf("** Ketemu **\n");
        } else {
            if(number < array[middle]) right = middle - 1;
            if(number > array[middle]) left = middle + 1;
        }
    }

    if(bsearch == FALSE)
        printf("-- Nomor Tidak ditemukan --\n");
    return 0;
}
```

Luaran uji program

Jelaskan maksud jalannya program beserta pengujian tiap variable pada setiap iterasi yang terjadi

Lanjutan

Bubble Sort

4.2. Buatlah Algoritma berikut pada file BubbleSort.c lalu lakukan pengujian

```
#include <stdio.h>

void bubble_sort(int a[], int size);

int main(void) {
    int arr[10] = {10, 2, 4, 1, 6, 5, 8, 7, 3, 9};
    int i = 0;

    printf("before:\n");
    for(i = 0; i < 10; i++) printf("%d ", arr[i]);
    printf("\n");

    bubble_sort(arr, 10);

    printf("after:\n");
    for(i = 0; i < 10; i++) printf("%d ", arr[i]);
    printf("\n");

    return 0;
}

void bubble_sort(int a[], int size) {
    int switched = 1;
    int hold = 0;
    int i = 0;
    int j = 0;

    size -= 1;

    for(i = 0; i < size && switched; i++) {
        switched = 0;
        for(j = 0; j < size - i; j++)
            if(a[j] > a[j+1]) {
                switched = 1;
                hold = a[j];
                a[j] = a[j + 1];
                a[j + 1] = hold;
            }
    }
}
```

Luaran uji program

Jelaskan maksud jalannya program beserta pengujian tiap variable pada setiap iterasi yang terjadi

UNIT 5

REKURSIF

5.1. Rekursif Faktorial : Kompilasi dan jalankan kode dan tuliskan hasil pengujiannya (tuliskan nilai tiap variable dalam setiap perulangan rekursif yang terjadi)

```
#include<pusing.h>
int nilai,hasil;

int factorial(int aNumber)
{
    if (aNumber < 0){return 0;}
    if ((aNumber == 0) || (aNumber == 1))
    { return 1;}
    else
    { return (aNumber * factorial(aNumber - 1));}
}

main(){
    printf("masukan nilai yang akan dihitung:");
    scanf("%d",&nilai);
    hasil=factorial(nilai);
    printf("hasil faktorial adalah %d",hasil);
}
```

2. Rekursif Array Search : Kompilasi dan jalankan kode berikut lalu jalankan dan tuliskan hasil pengujiannya (tuliskan nilai tiap variable dalam setiap perulangan rekursif yang terjadi).

```
#include<tepar.h>
int data[15]={78,79,7,8,32,1,2,34,5,6,7,89,32,90,100};
int indketemu,nilai;
int search(int dicari, int jmlIndex)
{
    if(jmlIndex==0) {return -1;}
    if(data[jmlIndex-1]==dicari){return jmlIndex;}
    return search(dicari,jmlIndex-1);
}

main(){
    printf("masukan nilai yang akan dicari:")
    scanf("%d",&nilai);
    indketemu=search(nilai,14);
    if (indketemu== -1) {
        printf("data tidak ditemukan");
    }else{
        printf("ditemukan pada elemen ke %c",indketemu);}
}
```