

### Uppgift nr 1

#### Ett fält:

Ett fält är ett sätt att skapa och hantera ett utrymme för ett antal variabler av samma typ.

Ex på deklaration av ett fält med plats för tio heltal:

```
int tabell[10];
```

#### Funktionsdeklaration:

Man deklarerar funktionens namn och egenskaper ( funktionshuvudet)

Ex : I fallet med en funktion enligt ovan gäller att man innan funktionen kan anropas i ett program har gjort en funktionsdeklaration enligt: `int funknamn(int tal_a);`

---

**En pekare** innehåller adressen till en variabel eller minnesutrymme. Pekaren definieras för att hantera en viss variabeltyp. Exempel på användning:

```
int *int_pekare;
int tabell[100];
pekare=tabell;
*pekare = 20; tabell[0] tilldelas värdet 20.
```

---

### Uppgift nr 2

#### Alt 1 :

```
#define ANTAL 3
int main(void){
    int tal[ANTAL];
    int min=INT_MAX;
    int n = 0;
    int smalest=min;
    while(1){
        printf("\n Mata in tre heltal : ");
        while (n < ANTAL) {
            printf("Ange tal %d: ", n+1);
            scanf("%d", &tal[n]);
            if ( smalest > tal[n]) smalest = tal[n];
            n++;
        }
        printf("Minsta talet var: %d\n\n", smalest);
        n=0;
    }
    return (0);
}
```

#### Alt 2 -----

```
int main(void){
    int a, b , c , min;
    while(1){
        printf("\n Mata in tre heltal : ");
        scanf( " %d %d %d",&a,&b,&c); // Andra typer av inmatning ok, Ej avdrag mellanslag mm
                                        som ställer till med problem I praktiken

        if((a<=b) && (a<=c)){
            min=a;
        } else if(b<=c) {
            min =b;
        } else min=c; // Logiken viktig del
    }
}
```

```
    printf("\n Min ar %d ", min);  
}  
system("PAUSE");  
return (0);  
}
```

---

### Uppgift nr 3

```
int main( void){  
    int x=-10;  
    printf("\nx\tf(x)");  
    printf("\n-----");  
    while( x< 11){  
        printf("\n %d\t%d",x,funk(x));  
        x++;  
    }  
    system("PAUSE");  
return(0);  
}
```

```
int funk(int x){  
    int res;  
    res = pow(x,3)-2*x*x+2*x-10;  
    return (res);  
}
```

---

### Uppgift 4

```
int main(void)  
{  
  
    int i;  
    while(1){  
        printf("\nAnge ett heltal (0-100) : ");  
        scanf("%d", &i);  
        if(i % 2 == 0) {  
            printf("\nTalet är jämnt");  
        } else {  
            printf("\nTalet är udda");  
        }  
    }  
    system("PAUSE");  
return 0;  
}
```

---

**Uppgift 5 a)**

```
int input(int list[], int val)
{
    int i, full=0;

    if( list[ MAX - 1 ] > 0 ){ /* Full */
        return (full);
    }else{
        i = MAX - 1;
        while( list[i] < 0 && i >= 0 ){
            i--;
        }
        list[i+1] = val;
        full=1;
    }
    return (full);
}
```

**5 b)**

```
void initQue( int list[] )
{
    int i;
    for( i = 0 ; i < MAX ; i++){
        list[i]=-1;
    }
}
```

---

**Uppgift nr 6**

```
#define ARRAYSIZE 100

int main () {
    int n = 0;
    char inText[ARRAYSIZE];
    printf("Ange tal: ");
    fgets(inText, ARRAYSIZE, stdin);
    while (inText[n] != '\0' || inText[n] != '\n' ) {
        if (isdigit(inText[n]) == 0) {
            break;
        }
        n++;
    }
    if (inText[n] == '\0' || inText[n] == '\n' ) {
        puts("Ok");
    }
    else {
        puts("Fel");
    }
    system("pause");
    exit(0);
}
```

---

### Uppgift nr 7

```
int SkrivutFilS( char *filnamn) {
    char rad[MAX];
    FILE *fil = fopen(filnamn, "r");
    if (fil == NULL) return 0;
    while((fgets(rad, MAX, fil)) != NULL) printf("%s", rad);
    return 1;
}
```

Alternativt :

```
int SkrivutFilC( char *filnamn) {
    char c;
    FILE *fil = fopen(filnamn, "r");
    if (fil == NULL) return 0;
    while((c=fgetc(fil)) != EOF) putchar(c);
    return 1;
}
```

---

### Uppgift nr 8

```
#define MAX 100
#define ANTAL 50
void skapa_slumptal( int tal[]);
int antal_primal(int tal[]);

int main( void){
    int tabell[ANTAL];
    int primal;
    skapa_slumptal(tabell);
    primal=antal_primal(tabell);
    printf("\n Antal primal %d \n", primal);
    for ( int a=0; a<ANTAL; a++) printf("%d ", tabell[a]);
    // anrop utskrift ; Skall ej skrivas av dig
    system("PAUSE");
    return(0);
}

// ----- Funktionsdefinitioner -----

void skapa_slumptal( int tal[]){
// Skapar antal st slumptal mellan 0 till max_tal till fältet enligt inparametern.
    srand(time(0));
    int n;
    for( n=0; n<ANTAL ; n++){
        tal[n]= (rand()% MAX)+1;
    }
}
```

```
//-----  
int antal_primal(int tal[]){  
    int n,div, primal,antal=0;  
    for(n=0; n<ANTAL; n++){  
        primal=1;  
        for(div = 2; div<tal[n];div ++){  
            if(div==tal[n]) continue;  
            if( tal[n] % div == 0 ) primal=0;  
        }  
        if(primal) antal++;  
    }  
    return(antal);  
}
```

---

Uppgift nr 9

```
void sortera( Posttyp post[], int max){  
    Posttyp temp;  
    int m,n,minpos;  
    for(m=0;m<max-1;m++){  
        minpos=m;  
        for( n=m+1;n<max;n++){  
            if(post[n].langd<post[minpos].langd){  
                minpos=n;  
            }  
        }  
        // Byt plats  
        if( minpos!=m){  
            temp=post[m];  
            post[m]=post[minpos];  
            post[minpos]=temp;  
        }  
    }  
}
```

---