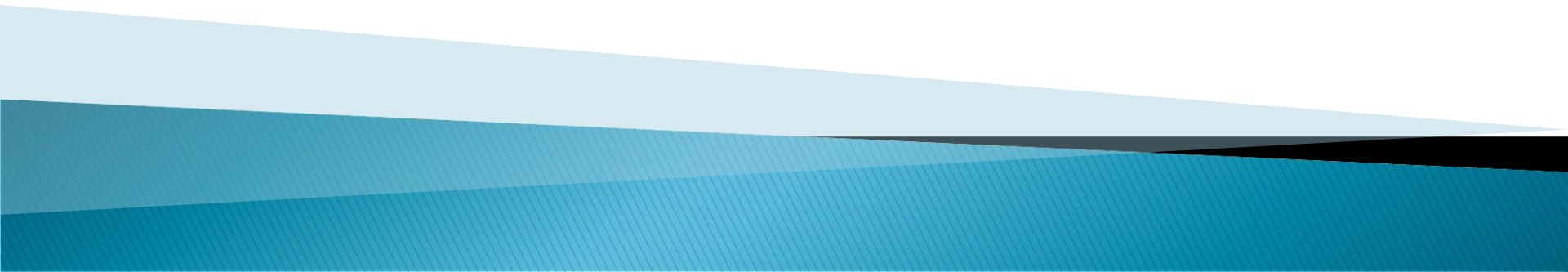


**Саркисян Г.Ф.**

# **ПРОГРАММИРОВАНИЕ**

**Часть I: Структурное программирование**



# АЛГОРИТМЫ СОРТИРОВКИ и ПОИСКА

## Вопросы:

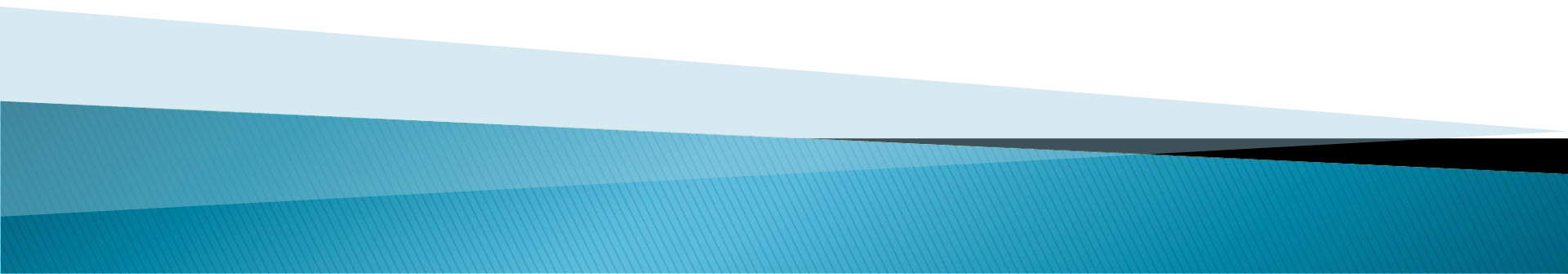
### *1. Алгоритмы сортировки*

- 1.1 Пузырьковый метод
- 1.2 Метод простого выбора
- 1.3 Метод простого включения
- 1.4.Метод Шелла
- 1.5 Метод Хоара
- 1.6 Сортировка слиянием

### *2. Алгоритмы поиска*

Бинарный поиск

# АЛГОРИТМЫ СОРТИРОВКИ



## Алгоритмы сортировки .ПУЗЫРЬКОВЫЙ МЕТОД

```
for( j = 1; j < n; ++j) {  
    for( i = 0; i < n - j ; ++i) {  
        if( x[i] > x[i+1]) {  
            buf = x[i];  
            x[i] = x[i+1];  
            x[i+1] = buf;  
        }  
    }  
}
```

## Алгоритмы сортировки . МЕТОД ПРОСТОГО ВЫБОРА

```
for( i = 0; i < n - 1; ++i) {  
    k = i;  
    for( j = i + 1; j < n; ++j)  
        if (x[k] > x[j])  
            k = j;  
    buf = x[i];  
    x[i] = x[k];  
    x[k] = buf;  
}
```

## Алгоритмы сортировки . МЕТОД ПРОСТОГО ВКЛЮЧЕНИЯ

```
for (i = 1; i < n; ++i) {  
    y = x[i];  
    j = i-1;  
    while (y < x[j] && j >= 0) {  
        x[j+1] = x[j];  
        j--;  
    }  
    x[j+1] = y;  
}
```

## Алгоритмы сортировки . МЕТОД ШЕЛЛА

```
for( k = n/2; k > 0; k /= 2) {  
    do {  
        k1 = 0;  
        for(i = 0, j = k; j < n; ++i, ++j)  
            if(x[i] > x[j]){  
                buf = x[i];  
                x[i] = x[j];  
                x[j] = buf;  
                ++k1;  
            }  
    }while(k1) ;  
}
```

## Алгоритмы сортировки . МЕТОД ХОАРА

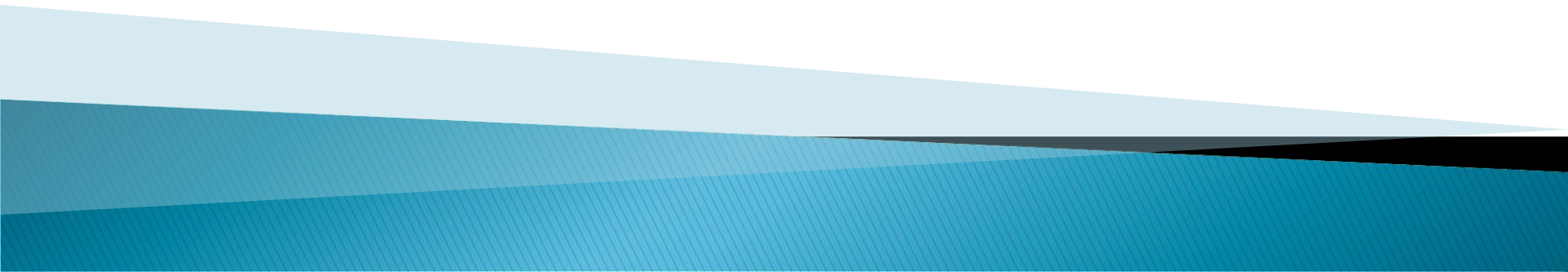
```
void sort_mas(int *a,int L,int R) {  
    int i = L, j = R;  
    int SR = a[(L+R)/2], buf;  
    do{  
        while(a[i] < SR) i++;  
        while(a[j] > SR) j--;  
        if(i <= j){  
            buf = a[i];        a[i] = a[j];  
            a[j] = buf;        i++;        j--; }  
    }while(i <= j);  
  
    if(i < R)    sort_mas(a, i, R);  
    if(j > L)    sort_mas(a, L, j);  
}
```



## Алгоритмы сортировки . СОРТИРОВКА СЛИЯНИЕМ

```
const int n1 = 5, n2 = 7, m = n1 + n2;  
double  arr1[n1] = {-90, -45, 67, 89, 100};  
double ptr[m] , arr2[n2] = {-34, -12, - 6,  
                             3, 7, 28, 200};  
  
int i1 = 0, i2 = 0, i3 = 0;  
while ((i1 < n1) && (i2 < n2))  
    if (arr1[i1] < arr2[i2] )  
        ptr[i3++] = arr1[i1++];  
    else ptr[i3++] = arr2[i2++];  
  
while(i1 < n1) ptr[i3++] = arr1[i1++];  
while(i2 < n2) ptr[i3++] = arr2[i2++];
```

# АЛГОРИТМЫ ПОИСКА



## Алгоритмы поиска. БИНАРНЫЙ ПОИСК

```
const int N1 = 6, N2 = 10;
```

```
int arr1[N1] = {45, -90, 78, 2, 67, -7};
```

```
int arr2[N2] = {-100, -90, -56, 2, 12, 45, 67,  
                78, 100, 598};
```

## Алгоритмы поиска. БИНАРНЫЙ ПОИСК

```
for(i = 0; i < N1; ++i){
    if(arr1[i] == arr2[N2/2]){
        printf("%d ",arr1[i]); continue;
    }
    if(arr1[i] > arr2[N2/2]){
        k = N2/2+1;  k1 = N2;
    }
    else{ k = 0;  k1 = N2/2;}

    for(j = k; j < k1; ++j)
        if(arr1[i] == arr2[j]){
            printf("%d ",arr1[i]); break;
        }
}
```

# СПАСИБО ЗА ВНИМАНИЕ!

## ВОПРОСЫ ?

### АЛГОРИТМЫ СОРТИРОВКИ и ПОИСКА

Автор: Саркисян Гаяне Феликсовна