

# ТЕМА 4. Алгоритмы поиска

# Линейный поиск

```
int p_lin1(tmas a[], int n, int x) {  
    for (int i = 0; i < n; i++)  
        if (a[i].key == x) return i;  
    return -1;  
}
```

# Линейный поиск с барьером

```
int p_lin2(tmas a[], int n, int x)
{
    a[n].key = x;
    int i = 0;
    while (a[i].key != x) i++;
return i;
}
```

# Поиск делением пополам

```
int p_dv(tmas a[], int n, int x) {  
    int i = 0, j = n - 1, m;  
    while (i < j) {  
        m = (i + j) / 2;  
        if (x > a[m].key) i = m + 1; else j = m;  
    }  
    if (a[i].key == x) return i;  
    return -1;  
}
```

# Интерполяционный поиск

```
int p_dv(tmas a[], int n, int x) {
    int i = 0, j = n - 1, m;
    while (i < j)
    {
        if (a[i].key == a[j].key) // исключение деления на ноль
            if (a[i].key == x) return i;
                else return -1;
        m = i + (j - i) * (x - a[i]) / (a[j] - a[i]);
        if (a[m].key == x) return m;
            else
                if (x > a[m].key) i = m + 1; else j = m - 1;
    }
    return -1; }
```