

```

template <typename T>
int binSearch(T x, T *a, int n)
{ int middle, left = 0, right = n-1;
  if (x < a[left]) return -1;
  if (x == a[left]) return 0;
  if (x > a[right]) return n;
  while (right-left > 1)
    { middle = (left + right)/2;
      if (x <= a[middle]) right = middle;
      else left = middle;
    }
  return right;
}

```

### Реализация на метода за файлове

Задача 67. Да се дефинира шаблон на функция *binSearch*, реализиращ двоично търсене на елемент във файл с пряка организация на достъп и с компоненти от тип *T*. Функцията *binSearch* да връща целочислена стойност *i*, дефинирана по начин, аналогичен на този от предходната задача.

```

template <typename T>
int binSearch(fstream& f, T x)
{ f.seekg(0, ios::end);

```

```
long n = f.tellg()/sizeof(T);
T al, ar, am;
int middle, left = 0, right = n-1;
f.seekg(left*sizeof(T));
f.read((char*)&al, sizeof(T));
f.seekg(right*sizeof(T));
f.read((char*)&ar, sizeof(T));
if (x < al) return -1;
if (x == al) return 0;
if (x > ar) return n;
while (right - left > 1)
    { middle = (left + right)/2;
      f.seekg(middle*sizeof(T));
      f.read((char*)&am, sizeof(T));
      if (x <= am) right = middle;
      else left = middle;
    }
return right;
}
```