内部结构：char数组+当前长的curlen

private char[] stringElem;

private int curlen;

部分功能实现：

1. public IString substring(int begin, int end) {//取出部分字符串

int j=0;

char[] nsqs = new char[end-begin];//需要新建一个数组

// for(int i=begin;i<end;i++){ //循环复制

// nsqs[j]=stringElem[i];

// j++;

// }

System.arraycopy(stringElem,begin,nsqs,0,end-begin);//用数据复制方法复制

return new SqString(nsqs);

}

2. public IString insert(int offset, IString str) { //插入

// if(str.length()+curlen<stringElem.length){

// int j=0;

// for(int i=offset;i<str.length();i++){

// char c=stringElem[i];

// stringElem[i]=str.charAt(j);

// stringElem[i+str.length()]=c;

// j++;

// }

// return new SqString(stringElem);

// }

// else{

// char[] nsqs = new char[curlen+str.length()];

// for(int i=0;i<offset;i++)

// nsqs[i]=stringElem[i];

// for(int i=offset,j=0;i<str.length()+offset;i++,j++)

// nsqs[i]=str.charAt(j);

// for(int i=offset;i<curlen;i++)

// nsqs[i+str.length()]=stringElem[i];

// return new SqString(nsqs);

// }

char[] nsqs = new char[curlen+str.length()];

char[] strarr = new char[str.length()];

for(int i=0;i<str.length();i++)

strarr[i]=str.charAt(i);

System.arraycopy(stringElem,0,nsqs,0,offset);

System.arraycopy(strarr,0,nsqs,offset,str.length());

System.arraycopy(stringElem,offset,nsqs,offset+str.length(),curlen-offset);

return new SqString(nsqs);

}

3. public IString delete(int beging, int end) { 删除

char[] nsqs = new char[curlen-(end-beging)]; //从新整理char数组

for(int i=0;i<beging;i++) //跳过要删除的数据复制到新的数组中

nsqs[i]=stringElem[i];

for(int i=end,j=beging;i<curlen;i++,j++)

nsqs[j]=stringElem[i];

return new SqString(nsqs);

}

4. public IString concat(IString str) { //连接，就等于在尾部插入字符串

return insert(curlen,str);

}

5. public int indexOf(IString str, int begin) { //子字符串的模式匹配

//有三个算法：BF、KMP、Sunday

}