

به نام خدا

گزارش پروژه دوم (باند، جیمز باند!) غزل پوراسفندیار بروجنی (9820453)

قرار است یک رشته حاوی حروف بزرگ انگلیسی را دریافت کرده و آن را به کلمات انگلیسی معنی دار، تفکیک کنیم.

برای این کار، متدهای زیر را نوشته ایم.

```
public static HashMap<String , Boolean> readDictionaryFromFile () {...}

public static boolean sentenceBreak(String sentence) {...}

public static boolean isValid(String word){...}

public static void reverseAndPrintResult(){...}
```

➤ در متد **readDictionaryFromFile** لغات مورد پذیرش در زبان انگلیسی را از فایل متنی با فرمت txt خوانده و در یک HashMap ذخیره می کنیم. فایل مربوط به لغات معنی دار حاوی حدود 370000 کلمه انگلیسی است که در بین آنها حروف مجزا هم، وجود دارد و اگر هر حرف الفبا به تنهایی یک کلمه با معنا باشد هر رشته ای به حروف تشکیل دهنده ای تقسیم می شود و در نتیجه همه رشته ها با معنی تلقی می شوند در حالی که ممکن است رشته ای بی معنا مثل "ABCDEFGH" بوده باشد. به همین دلیل حروف الفبای انگلیسی به جز "a" و "I" که در انگلیسی دارای معنی هستند را با value = false ذخیره کردیم.

➤ متد **isValid** یک کلمه را به عنوان ورودی دریافت می کند و اگر در دیکشنری وجود داشت و مقدار value آن هم برابر true بود، معتبر بودن این کلمه را به صورت Boolean بر می گرداند.

➤ در متد **sentenceBreak** الگوریتم شکستن جمله به کلمات به صورت بازگشتی انجام می شود. شرط های پایه این است که یا طول جمله ورودی صفر باشد و یا جمله ورودی در دیکشنری معتبر باشد. در غیر این صورت، جمله را به دو بخش تقسیم می کند و دوباره بررسی می کند که آیا قسمت های به وجود آمده معتبر هستند یا نه. طبق نحوه نوشتن الگوریتم، اولین بار به آخرین کلمه معنا دار در جمله می رسیم. آن را در لیستی ذخیره می کنیم و در آخر در متد **reverseAndPrintResult** لیست نهایی برعکس شده و چاپ می گردد. برای مثال اگر جمله ما "THISISME" باشد، نحوه کارکرد الگوریتم بعد از تبدیل جمله به حروف کوچک، به صورت زیر است: (کلمات با معنا به رنگ سبز نشان داده می شوند).

sentenceBreak("thisisme")

thisisme : thisism + e

thisim : thisis + me

thisis : thisi + sme

thisi : **this** + isme

(در اینجا همچنان متد صدا زده می شود و "this" در اینجا در لیست ذخیره نمی شود.)

now call sentenceBreak("isme")

isme : ism + e

ism : **is** + me

now call sentenceBreak("me")

me : **me** + ""

و حال در اینجا ذخیره سازی در لیست از آخر به اول شروع می شود و به همین دلیل آنها را در لیست برعکس ذخیره می شوند.

مثالی از ورودی برنامه :

```
sentenceBreak("HELLOEVERYBODYTHISISME"); //hello everybody this is me
reverseAndPrintResult();

sentenceBreak("YOUCANLEARNENGLISHLANGUAGE"); //you can learn english language
reverseAndPrintResult();

sentenceBreak("TREEISSTREET"); //tree is street
reverseAndPrintResult();

sentenceBreak("THEREISATREEINTHESTREET"); //there is a tree in the street
reverseAndPrintResult();

sentenceBreak("CALLSECURITYATHIAMIAIRPORTBECAUSEITHINKABOMBISABOUTTOGOOOF");
reverseAndPrintResult();
```

مثالی از خروجی برنامه :

قبل از دیدن خروجی برای جملات مختلف باید دقت داشته باشیم که به علت حجم عظیم کلمات تعریف شده در دیکشنری، ممکن است خروجی آنچه که انتظار می رود نباشد اما همچنان جمله به لغات قابل تعریف در دیکشنری تبدیل شده است (دو مثال آخر در خروجی ها).

```
hello everybody this is me  
you can learn english language  
tree is street  
there is at ree int hest reet  
calls ecu rit yat miamia ir port because it hin kab om bis about togo off
```