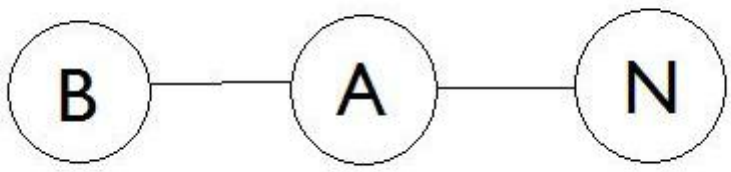
June Training Graph Theory Contest

Problem 2: *parkpaths*

Problem Description

Arnold is considering building a memorial park to commemorate his best friend. After considering for a long time, Arnold has thought of a creative way to do so. The memorial park will contain many different fountains, each labelled with a uppercase letter. Note that there can be many fountains with the same letter. Besides that, some pairs of fountains can also be connected with a gravel footpath. Also take note that gravel footpaths can lead to themselves.

Arnold wants to build the memorial park such that the name of his best friend *S* forms a path in the memorial park, but also wishes to minimise the amount of time needed to build the memorial park. Each fountain takes 2 units of time to build whereas each footpath takes 1 unit of time to build. For example, imagine Arnold had a friend named "BANANA". The best way to build this memorial park is as follows.



A path that forms "BANANA" is from the leftmost "B" to the middle "A" then back and forth "A" and "N" twice. This park has three fountains and two footpaths. The minimum time required to build this memorial park is thus 8. Given Arnold's best friend's name, help find out the minimum time required to build a memorial park such that the name forms a path in the park.

Implementation

Your program is required to include the header file `parkpaths.h` as part of your code.

In addition, your program will be required to implement the functions:

- `int parkpaths(string S)`, which returns the minimum time required to build such a memorial park.

Please **do not** input or output anything to or from standard input (stdin) or standard output (stdout). Submissions that do so will be graded as **Incorrect**.

Grading

All submissions should adhere to the following time and memory constraints:

- Time Limit: 1 second.
- Memory Limit: 64 Megabytes.

All testdata for this problem will satisfy the following conditions:

- All letters in S are uppercase alphabetical characters.
- $1 \leq |S| \leq 100\,000$.

In addition, partial credit can also be obtained by passing **all** testdata from the subtasks:

Subtask 1 (7 points)

- $1 \leq |S| \leq 5$.

Subtask 2 (15 points)

- $1 \leq |S| \leq 100$.

Subtask 3 (32 points)

- $1 \leq |S| \leq 3\,000$.

Subtask 4 (46 points)

- $1 \leq |S| \leq 100\,000$.

Testing

In order to help you in testing your submission, *grader.cpp* will be provided. A starting template *parkpaths.cpp* will also be provided. It should be compiled together with your submission and run with the sample inputs to test your program before submission.

Sample Input 1

```
BANANA
```

Sample Output 1

```
8
```

Sample Input 2

```
SAMUEL
```

Sample Output 2

```
17
```