# Problem 3 – AJAX Request Validator

You will be given several AJAX requests which will be in a very custom format:

**“Method: {Method}**

**Credentials: {Authentication token}**

**Content: {Content}”**

Those elements are each on a different line. The valid method can be “**GET**”, “**POST**”, “**PUT**”, “**DELETE**”. The valid authentication token must be in format “**Basic asdfg123…”** or **“Bearer asdfg123…”**. In other words, the authentication token must contain the authorization type (**Basic** or **Bearer**), a **single** space, and an authentication string of **one or more** characters, which can be - English alphabet letters and digits. The content can contain letters, digits and **dots**. There is a single space between the element caption and its value. Any input request that doesn’t follow the, specified above - format, is to be treated as **invalid**.

On the final line you will get **a hash pattern**, with which you must **decode** the **authorization token** in order to print it in the **success response**. The hash pattern will consist of blocks of a **digit** and a **letter**. For example: **“2s3b4a”**. For the authorization token to be **valid** it must contain at least one of those letters, **exactly**, **equal to the digit in front of it**, **amount** of times. If the authorization token does not contain any of the letters, **exactly its** specified times, the request becomes **invalid**.

To perform a **GET** request you can have **Basic** authorization, but for every other your need **Bearer**. If request succeeds you must print a **success response**: **“Response–Method:{method}&Code:200&Header:{requestHeader}”**

If a request **fails** because of **invalid format**, (**NOT** in the format, specified above) the failure response should be:  
“**Response-Code:400”**

If a request fails because of **invalid authentication type**, (**Basic** on a **POST /** **PUT / DELETE**) the response should be: **“Response–Method:{method}&Code:401“**

If a request fails because of **invalid authorization token**, (**Hash password** failed to decode) the response should be: **“Response–Method:{method}&Code:403“**

In the response strings “method” is the **current request’s method** and “request header” is the **authorization token** in the **credentials** of the current request, after the authorization type.

### Input

* The input data is passed to the first JavaScript function found in your code as an **array of strings**.
* On several lines of input you will receive the requests.
* On the final line you will receive the decoding hash for the authentication string.

### Output

* The output should consist only of responses, which must be in the format specified above.

### Constraints

* The amount of input requests will be in range [0, 1000].
* The hash pattern will consist **only** of blocks of digits, and English alphabet letters **after** them.
* It is safe to assume there will be no requests with missing elements.
* Allowed time/memory: 250ms/16MB

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| Method: GET  Credentials: Bearer asd918721jsdbhjslkfqwkqiuwjoxXJIdahefJAB  Content: users.asd.1782452.278asd Method: POST  Credentials: Basic 028591u3jtndkgwndsdkfjwelfqkjwporjqebhas  Content: Johnathan  2q | Response‑Method:GET&Code:200&Header:asd918721jsdbhjslkfqwkqiuwjoxXJIdahefJAB  Response-Method:POST&Code:401 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| Method: PUT  Credentials: Bearer as9133jsdbhjslkfqwkqiuwjoxXJIdahefJAB  Content: users.asd/1782452$278///\*\*asd123  Method: POST  Credentials: Bearer 028591u3jtndkgwndskfjwelfqkjwporjqebhas  Content: Johnathan  Method: DELETE  Credentials: Bearer 05366u3jtndkgwndssfsfgeryerrrrrryjihvx  Content: This.is.a.sample.content  2e5g | Response-Code:400  Response‑Method:POST&Code:200&Header:028591u3jtndkgwndskfjwelfqkjwporjqebhas  Response‑Method:DELETE&Code:200&Header:05366u3jtndkgwndssfsfgeryerrrrrryjihvx |