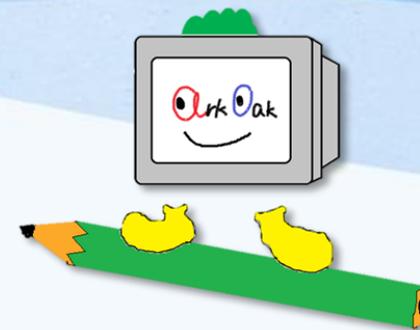
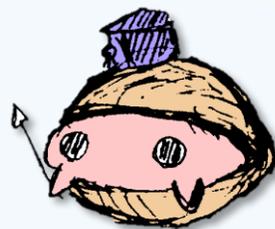


アルゴリズム勉強会 第3回



文字列の復習

```
#include <stdio.h>
```

```
int main(void)  
{  
    char c[] = "Bye";  
    printf("%s", c);  
    return 0;  
}
```

文字列はダブルクォーテーション

文字列の表示には %s

```
#include <stdio.h>
```

```
int main(void)  
{  
    char c[4];  
    c[0] = 'B';  
    c[1] = 'y';  
    c[2] = 'e';  
    c[3] = '¥0';  
    printf("%s", c);  
    return 0;  
}
```

文字定数はシングルクォーテーション

文字の終わりを表すヌル文字 '¥0'

文字列は、文字定数(char)の配列だったね



文字列の取得

```
#include <stdio.h>
```

```
int main(void)
```

```
{
```

```
    char c[64];
```

```
    scanf("%s", &c);
```

```
    printf("%s", c);
```

```
    return 0;
```

```
}
```

十分な大きさの配列を確保しておく

scanf() で他の変数同様に取得可能

gets() もあったね



cctype.h

関数名	役割
isalpha	文字定数がアルファベットかどうかチェック
islower	文字定数がアルファベットの小文字かどうかチェック
isupper	文字定数がアルファベットの大文字かどうかチェック
isdigit	文字定数が10進数の数字かどうかをチェック

関数名	役割
tolower	アルファベットの大文字を小文字へ変換する
toupper	アルファベットの小文字を大文字へ変換する



大文字と小文字を入れ替えよう

```
#include <stdio.h>
#include <ctype.h>

int main(void)
{
    int i;
    char c[64];
    scanf("%s", &c);
    for (i = 0; c[i] != '\0'; i++){
        if (islower(c[i])){
            c[i] = toupper(c[i]);
        } else if (isupper(c[i])){
            c[i] = tolower(c[i]);
        }
    }
    printf("%s", c);
    return 0;
}
```

ctype.h のインクルード

文字列の終わり ('¥0') が来るまで繰り返す

c[i] が小文字かどうかチェック
小文字だった場合、大文字に変換

c[i] が大文字かどうかチェック
大文字だった場合、小文字に変換

実行例

```
HelloWorld
hELLOwORLD
```



281 A. Word Capitalization

大文字化は、単語の初めの文字を大文字で書くことです。

あなたの仕事は与えられた単語を大文字化することです。

※大文字化では、単語の初めの文字以外は変わりません。

入力

一行の空でない文字列が入力されます。

この文字列は、大文字と小文字を含むアルファベットです。

文字列の長さは1000を超えません。

出力

大文字化を行った文字列を出力して下さい。

配列の一番はじめを大文字にするだけ!



281A. 解答例

```
#include <stdio.h>
#include <ctype.h>

int main(void)
{
    char c;
    scanf("%c", &c);
    c[0] = toupper(c[0]);
    printf("%c", c);
    return 0;
}
```

1000文字読み込めるように準備

一文字目を大文字に変換

さっきの練習問題より簡単



118A. String Task

ペチャはプログラミング教室に参加しました。

彼の最初の作業は、簡単なプログラムを書くことでした。

そのプログラムは、与えられた文字列に対して次の処理を行います：

- 全ての母音を削除します。
- 全ての子音の前に"."を挿入します。
- 全ての大文字の子音を、小文字に置き換えます。

母音は、"A", "O", "Y", "E", "U", "I", であり、残りの文字は子音です。

プログラムの入力文字列で与えられ、処理を施した結果の文字列を出力します。

ペチャがこの簡単な作業をするのを手伝って下さい。

処理の順番を考えよう



118A. 解答例

```
#include <stdio.h>
#include <ctype.h>
```

```
int main(void)
```

```
{
```

```
    int i;
```

```
    char c[101];
```

```
    scanf("%s", &c);
```

```
    for (i = 0; c[i] != '\0'; i++) {
```

```
        c[i] = tolower(c[i]);
```

```
        if (!(c[i] == 'a' || c[i] == 'e' || c[i] == 'i' ||
```

```
              c[i] == 'o' || c[i] == 'u' || c[i] == 'y' ||
```

```
              c[i] == 'e' || c[i] == 'i')) {
```

```
            printf("%.1s", c[i]);
```

```
        }
```

```
    }
    return 0;
```

```
}
```

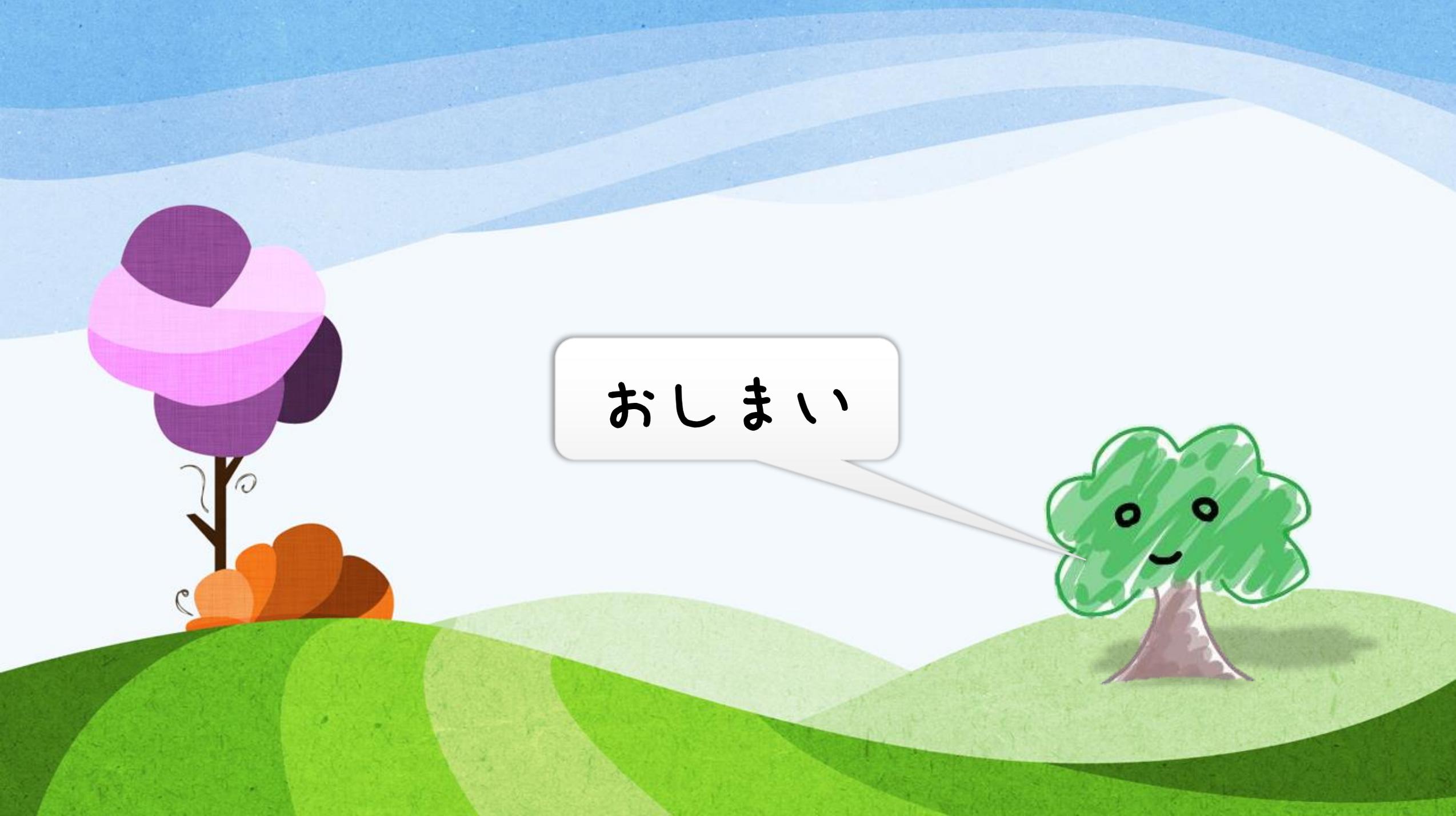
全ての文字を小文字に変換する

母音か子音かをチェック

子音ならば、前に '.' をつけてprintf する

最後にまとめてprintfする必要はないんだ





おしまい