
Прямой

```
In[ ]:= space[str_] := Module[{strEx = StringPartition[str, 1], lst = {}, lstWord = {},
  sPos = 1, wordForLst = 0, k = 0, indx = 0},
  While[sPos ≤ StringLength@str,
    For[j = sPos, j ≤ StringLength@str, j++,
      word = StringJoin@strEx[[sPos ;; j]];
      AppendTo[lstWord, word]];
    For[i = 1, i ≤ Length@lstWord, i++,
      If[DictionaryLookup[lstWord[[i]]] ≠ {},
        If[StringLength@lstWord[[i]] > k, k = StringLength@lstWord[[i]];
          indx = i]]];
    If[indx ≠ 0,
      AppendTo[lst, lstWord[[indx]]];
      sPos += k, AppendTo[lst, strEx[[sPos]]]; sPos++];
    indx = 0;
    k = 0;
    lstWord = {};
    StringJoin@Riffle[lst, " "]]

str = "themendinehere";
space[str]
```

Out[]:= theme n dine here

Обратный

```
In[ ]:= space2[str_] :=
  Module[{strEx = Reverse@StringPartition[str, 1], lst = {}, lstWord = {},
    sPos = 1, wordForLst = 0, k = 0, indx = 0},
    While[sPos ≤ StringLength@str,
      For[j = sPos, j ≤ StringLength@str, j++,
        word = StringJoin@Reverse@strEx[[sPos ;; j]];
        AppendTo[lstWord, word]];
      For[i = 1, i ≤ Length@lstWord, i++,
        If[DictionaryLookup[lstWord[[i]]] ≠ {},
          If[StringLength@lstWord[[i]] > k, k = StringLength@lstWord[[i]];
            indx = i]]];
      If[indx ≠ 0,
        AppendTo[lst, lstWord[[indx]]];
        sPos += k, AppendTo[lst, strEx[[sPos]]]; sPos++];
      indx = 0;
      k = 0;
      lstWord = {};
      StringJoin@Riffle[Reverse@lst, " "]]

str = "themendinehere";
space2[str]
```

Out[]:= the men dine here

Двунаправленный

```
In[ ]:= space3[str_] := Module[
  {num1 = 0, num2 = 0, str1 = StringSplit@space[str], str2 = StringSplit@space2[str]},
  For[i = 1, i ≤ Length@str1, i++,
    If[DictionaryLookup[str1[[i]]] ≠ {}, num1 += 1]];
  For[i = 1, i ≤ Length@str2, i++,
    If[DictionaryLookup[str2[[i]]] ≠ {}, num2 += 1]];
  If[num2 > num1, space2[str], space[str]]]

str = "themendinehere";
space3[str]

Out[ ]:= the men dine here
```

‘Рекурсивный’

```
In[ ]:= space4[start_, str_] :=
  Module[{strEx = StringPartition[str, 1], lst = {}, lstWord = {},
    sPos = 1, wordForLst = 0, k = 0, indx = 0},
  For[j = sPos, j ≤ StringLength@str, j++,
    word = StringJoin@strEx[[sPos ;; j]];
    If[DictionaryLookup[word] ≠ {},
      AppendTo[lstWord, word]]]; Flatten@{start, #} & /@ lstWord]

space4main[str_] := Module[{strNew = str, start, k = 1, l, var = {}},
  start = space4[{}, strNew];
  While[start ≠ {},
    For[i = 1, i ≤ Length@start, i++,
      If[StringLength@StringJoin@start[[i]] == StringLength@str, Nothing,
        l = start[[i]];
        start[[i]] = space4[start[[i]], StringJoin[
          StringPartition[strNew, 1][StringLength@StringJoin@start[[i]] + 1 ;;]]];
      start = Flatten[start, 1];
      For[i = 1, i ≤ Length@start, i++,
        If[StringLength@StringJoin@start[[i]] == 14, AppendTo[var, start[[i]];
          start[[i]] = {}]];
      start = DeleteCases[start, {}]];
  {StringJoin@Riffle[var[[1]], " "], var}]

str = "themendinehere";
space4main[str]

Out[ ]:= {the men dine here, {{the, men, dine, here}, {them, en, dine, here},
  {the, men, din, eh, ere}, {the, men, dine, he, re}, {the, mend, in, eh, ere},
  {them, en, din, eh, ere}, {them, en, dine, he, re}, {them, end, in, eh, ere}}}
```