

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Сибирский государственный университет телекоммуникаций и информатики»
(СибГУТИ)

Кафедра прикладной математики и кибернетики

РАСЧЕТНО-ГРАФИЧЕСКОЕ ЗАДАНИЕ
по дисциплине «Функциональное и логическое программирование»

Вариант 16

Выполнил:

студент группы ИП-811
Разумов Д.Б.

Работу проверил:

Галкина М.Ю.

Оглавление

Задания.....	3
Листинг.....	4
Скриншоты.....	7

Задания

1. Вставьте в список новый элемент X перед всеми вхождениями Y, если Y входит в исходный список. Например, [1,2,1,5,8,1,0], X=5, Y=1 -> [5,1,2,5,1,5,8,5,1,0].
2. Из текстового файла, содержащего несколько строк, удалите все слова максимальной длины. Сформируйте новый файл.

Листинг

% ЗАДАНИЕ 1.16

task1([H1|T1], X, Y, [X,H1|T2]):-

 H1 == Y, !, task1(T1, X, Y, T2).

task1([H1|T1], X, Y, [H1|T2]):-

 task1(T1, X, Y, T2).

task1([], _, _, []).

% task1([1,2,1,5,8,1,0], 5, 1, List). % последние эл-ты не выводятся

% task1([1,5,8,1,0], 5, 1, List). % а здесь выводится

% task1([5,4,3,2,1], 3, 2, List).

% ЗАДАНИЕ 2.16

task2():-

 open('E:\\programming\\FLP\\rgz\\in.txt', read, F),

 set_input(F), read_stream_to_codes(F, List), close(F),

 isolate_nl(List, NewList), % выделить nl как отдельное слово

 N is 0, TempMaxN is 0, find_max_length(NewList, N, TempMaxN, MaxN),

 !, write('max length = '), writeln(MaxN),

 codes_to_words(NewList, Words), %write(Words),

 tell('E:\\programming\\FLP\\rgz\\out.txt'),

 write_to_file(Words, MaxN), told.

% выделение символа перевода строка пробелами:

isolate_nl([Head|Tail1], [Head|Tail2]):-

 Head \\= 10, !, isolate_nl(Tail1, Tail2).

isolate_nl([10|Tail1], [32,10,32|Tail2]):-

isolate_nl(Tail1, Tail2).

isolate_nl([], []).

% найти длину самого длинного слова:

find_max_length([Head|Tail], N, MaxN, Ans):-

Head \== 32, Head \== 10, !, N1 is N + 1,

find_max_length(Tail, N1, MaxN, Ans).

find_max_length([_|Tail], N, MaxN, Ans):-

(N @> MaxN, MaxN1 is N;

MaxN1 is MaxN),

N1 is 0, find_max_length(Tail, N1, MaxN1, Ans).

find_max_length([], N, MaxN, Ans):-

(N @> MaxN, Ans is N; % последнее слово может быть максимальным

Ans is MaxN), !.

% убрать пробелы перед словом:

get_spaces([Head|Tail], [Head|TailSpaces], TailString):-

Head == 32, !, get_spaces(Tail, TailSpaces, TailString).

get_spaces(List, [], List).

% выделить слово:

get_word([Head|Tail], [Head|TailSpaces], TailString):-

Head \== 32, !, get_word(Tail, TailSpaces, TailString).

get_word(List, [], List).

% выделяет первое слово из строки (в начале могут идти пробелы)

read_word(String, Word, Tail):-

```

get_spaces(String, _, StringWithoutSpaces), % убирает пробелы

get_word(StringWithoutSpaces, Word, Tail),

Word \== [].

```

% преобразовать коды символов в слова:

```

codes_to_words(String, [Word|ListTail]):-

```

```

    read_word(String, Word, StringTail), !,

```

```

    codes_to_words(StringTail, ListTail).

```

```

codes_to_words(_, []).

```

%запись слов в файл:

```

write_to_file([Head|Tail], Ans):-

```

```

    length(Head, LenHead), LenHead == Ans, !, write_to_file(Tail, Ans).

```

```

write_to_file([Head|Tail], Ans):-

```

```

    string_to_list(S,Head), write(S),

```

```

    check_nl(Head),

```

```

    write_to_file(Tail, Ans).

```

```

write_to_file([], _).

```

%если слово - не перевод строки, то добавить после него пробел:

```

check_nl([Head|_]):-

```

```

    Head \== 10, !, write(' ').

```

```

check_nl(_).

```

Скриншоты

?- task1([1,2,1,5,8,1,0], 5, 1, List).

List = [5, 1, 2, 5, 1, 5, 8, 5, 1|...].

?- task1([1,5,8,1,0], 5, 1, List).

List = [5, 1, 5, 8, 5, 1, 0].

?- task1([5,4,3,2,1], 3, 2, List).

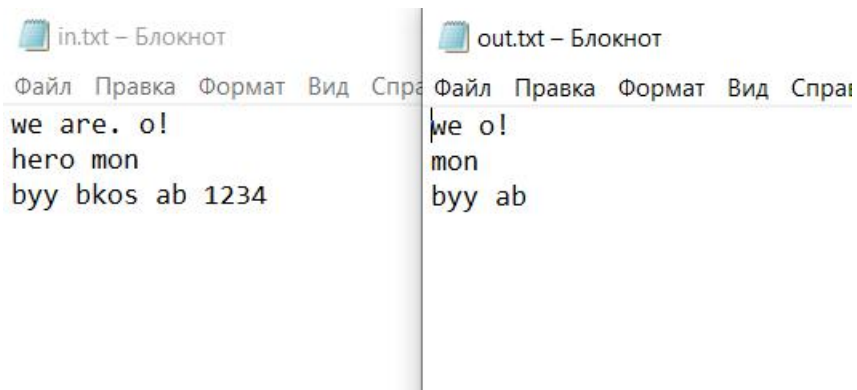
List = [5, 4, 3, 3, 2, 1].

?- task1([5,4,3,2,1], 3, 12, List).

List = [5, 4, 3, 2, 1].

?-

Скриншот 1 - выполнение задания 1



Скриншот 2 - выполнение задания 2